

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:

Kazuhiko IKEUCHI et al.

Application No.: To be Assigned

Group Art Unit: To be Assigned

Filed: November 28, 2001

Examiner: To be Assigned

For: RAID CONTROLLER AND CONTROL METHOD THEREOF

**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN  
APPLICATION IN ACCORDANCE  
WITH THE REQUIREMENTS OF 37 C.F.R. § 1.55**

Assistant Commissioner for Patents  
Washington, D.C. 20231

Sir:

In accordance with the provisions of 37 C.F.R. § 1.55, the applicant(s) submit(s) herewith a certified copy of the following foreign application:

Japanese Patent Application No. 2001-220179

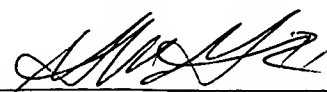
Filed: July 19, 2001

It is respectfully requested that the applicant(s) be given the benefit of the foreign filing date(s) as evidenced by the certified papers attached hereto, in accordance with the requirements of 35 U.S.C. § 119.

Respectfully submitted,

STAAS & HALSEY LLP

Date: Nov 28, 2001

By:   
Gene M. Garner II  
Registration No. 34,172

700 11th Street, N.W., Ste. 500  
Washington, D.C. 20001  
(202) 434-1500

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

1c872 U.S. PTO  
09/994698  
11/28/01

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2001年 7月19日

出 願 番 号

Application Number:

特願2001-220179

出 願 人

Applicant(s):

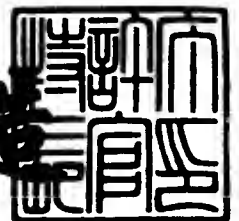
富士通株式会社

CERTIFIED COPY OF  
PRIORITY DOCUMENT

2001年 9月 5日

特 許 庁 長 官  
Commissioner,  
Japan Patent Office

及 川 耕 造



出証番号 出証特2001-3082306

【書類名】 特許願

【整理番号】 0151662

【提出日】 平成13年 7月19日

【あて先】 特許庁長官 及川 耕造 殿

【国際特許分類】 G06F 3/06  
G11B 19/02

【発明の名称】 R A I D制御装置及びその制御方法

【請求項の数】 5

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士  
通株式会社内

【氏名】 池内 和彦

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士  
通株式会社内

【氏名】 伊藤 実希夫

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士  
通株式会社内

【氏名】 大黒谷 秀治郎

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士  
通株式会社内

【氏名】 間宮 小登美

【発明者】

【住所又は居所】 神奈川県横浜市港北区新横浜2丁目4番19号 株式会  
社富士通プログラム技研内

【氏名】 北村 嘉朗

【発明者】

【住所又は居所】 神奈川県横浜市港北区新横浜 2 丁目 4 番 1 9 号 株式会  
社富士通プログラム技研内

【氏名】 村松 実

【発明者】

【住所又は居所】 神奈川県横浜市港北区新横浜 2 丁目 4 番 1 9 号 株式会  
社富士通プログラム技研内

【氏名】 星野 弘行

【特許出願人】

【識別番号】 000005223

【氏名又は名称】 富士通株式会社

【代理人】

【識別番号】 100094514

【弁理士】

【氏名又は名称】 林 恒▲徳▼

【代理人】

【識別番号】 100094525

【弁理士】

【氏名又は名称】 土井 健二

【手数料の表示】

【予納台帳番号】 030708

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9704944

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 R A I D 制御装置及びその制御方法

【特許請求の範囲】

【請求項 1】 上位装置からのディスクアクセス要求に応じて、複数の物理ディスクに分散して格納された論理ボリューム上のアクセス要求領域をアクセスする R A I D 制御装置において、

各々異なる論理ボリュームを担当し、1つの論理ボリュームを複数の物理ディスクに分散して、冗長グループを組む複数の物理ディスク群と、

前記論理ボリュームの論理フォーマット処理の進捗情報を管理するための管理テーブルを参照して、前記論理ボリュームの各領域の論理フォーマット要求を発行する複数の論理フォーマット管理モジュールと、

前記論理フォーマット管理モジュールの処理要求に応じて、前記物理ディスクを実アクセスする複数の下位層モジュールとを有し、

前記各論理フォーマット管理モジュールは、

前記複数の物理ディスク群を担当するため、前記各物理ディスク群の管理テーブルを有し、

前記ディスクアクセス要求に応じて、前記管理テーブルを参照し、当該アクセス領域全てが、論理フォーマット済かを判定し、フォーマット済と判断した場合に、前記下位層モジュールにディスクアクセスを要求し、フォーマット済でなければ、論理フォーマット処理要求を発行し、前記ディスクアクセス要求を待ちキューにキューイングし、前記待ちキューにディスクアクセス要求がない時は、前記管理テーブルから未フォーマット領域を検索して、論理フォーマット処理要求を発行することを

特徴とする R A I D 制御装置。

【請求項 2】 一の前記論理フォーマット管理モジュールの異常に応じて、他の前記論理フォーマット管理モジュールで、前記一の論理フォーマット管理モジュールの担当する物理ディスク群の論理フォーマットを実行することを

特徴とする請求項 1 の R A I D 制御装置。

【請求項 3】 一の前記論理フォーマット管理モジュールの異常に応じて、他

の前記論理フォーマット管理モジュールの管理テーブルから、前記一の論理フォーマット管理モジュールの担当する物理ディスク群の管理テーブルを、前記一論理フォーマット管理モジュール以外の他の論理フォーマット管理モジュールに復元する R A I D 管理モジュールを有することを

特徴とする請求項 1 の R A I D 制御装置。

【請求項 4】前記 R A I D 管理モジュールは、R A I D 構成を記述した構成テーブルを参照し、前記復元を実行することを

特徴とする請求項 3 の R A I D 制御装置。

【請求項 5】上位装置からのディスクアクセス要求に応じて、各々異なる論理ボリュームを担当し、1つの論理ボリュームを複数の物理ディスクに分散して、冗長グループを組む複数の物理ディスク群のアクセス要求領域をアクセスする R A I D 制御方法において、

前記論理ボリュームの論理フォーマット処理の進捗情報を管理するための管理テーブルを参照して、前記論理ボリュームの各領域の論理フォーマット要求を発行する論理フォーマット要求ステップと、

前記ディスクアクセス要求に応じて、前記管理テーブルを参照し、当該アクセス領域全てが、論理フォーマット済かを判定するステップと、

フォーマット済と判断した場合に、前記物理ディスクを実アクセスする下位層モジュールにディスクアクセスを要求するステップと、

フォーマット済でなければ、前記論理フォーマット処理要求を発行し、前記ディスクアクセス要求を待ちキューにキューイングするステップと、

前記下位層モジュールからの前記論理フォーマット処理の終了通知に応じて、冗長性を持つ前記管理テーブルを更新するステップとを有することを

特徴とする R A I D 制御方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、磁気ディスク等の物理ディスクを使用してデータを管理するストレージ制御装置及びその方法に関し、特に、データに冗長性を持たして管理する R

A I D 制御装置及びその制御方法に関する。

【 0 0 0 2 】

【従来の技術】

磁気ディスク、光磁気ディスク、光ディスク等の記憶媒体を利用したストレージ機器では、データ処理装置の要求で、記憶媒体を実アクセスするため、記憶媒体にフォーマットの書き込みが必要である。このフォーマットとして、CKDフォーマット等の物理フォーマットと、データ領域をホストOSの処理単位（例えば、ブロック単位）に区切り、識別を施す論理フォーマットとがある。

【 0 0 0 3 】

このような記憶媒体のフォーマット化により、ホストOSのホストコマンドにより、記憶媒体のアクセス（リード／ライト等）が可能となる。物理フォーマットは、媒体のトラック単位の初期化处理であり、論理フォーマットは、物理フォーマットされたトラックのデータ領域のブロック単位のフォーマット処理を意味する。この内、フォーマット処理は、媒体の全てのブロックをフォーマットするには、時間がかかり、フォーマットされていない媒体を、実際に使用するには、長時間（例えば、30分以上）のフォーマット処理を待つ必要があった。

【 0 0 0 4 】

このため、記憶媒体のフォーマット処理を初期化处理として行わずに、直ちに使用可能とし、ライトアクセスを受けた時に、その対象ブロックをフォーマット処理し、ライトする方法が提案されている（例えば、特開平8-147110号公報等）。

【 0 0 0 5 】

【発明が解決しようとする課題】

従来技術では、アクセスを受けた時に、論理フォーマットを実行するため、アクセスを受けない場合には、論理フォーマットを実行しないものであった。この従来技術は、MD（ミニディスク）のように、1枚の媒体に、音楽、画像等を連続的にライトする場合には、好適である。

【 0 0 0 6 】

しかしながら、ホストやサーバー等に接続される大規模ストレージシステムの

ように、ランダムにアクセスがある場合には、アクセス毎に、論理フォーマット処理及びアクセス処理を実行する必要があるため、アクセス処理に要する時間が遅くなり、アクセス時間の長期化を招くという問題が生じる。

## 【0007】

特に、データに冗長性を持たせて管理するRAIDストレージシステムでは、論理フォーマットに要する時間が、論理ボリュームの大きさにより異なるが、およそ数十分から数時間かかるため、ストレージシステムの新設や、既存システムに論理ボリュームを増設するのに、アクセス時間が長くなるという問題が生じる。

## 【0008】

更に、RAIDシステムでは、データの冗長性を考慮した論理フォーマット化を必要とする。

## 【0009】

従って、本発明の目的は、冗長性を維持しつつ、論理フォーマットを実行しながら、ホストアクセスを含むデータ処理を実行するためのRAID制御装置及び制御方法を提供するにある。

## 【0010】

又、本発明の他の目的は、論理フォーマット中に、多重化ユニットの一のユニットが故障しても、リビルド時に、論理フォーマットの実行の冗長構成に復元するためのRAID制御装置及びその制御方法を提供するにある。

## 【0011】

更に、本発明の他の目的は、冗長構成を維持しつつ、ホストアクセスに伴う論理フォーマット処理と、ホストアクセスに関係しない論理フォーマット処理を並列に実行して、ホストアクセス処理の遅れを短縮するためのRAID制御装置及びその制御方法を提供するにある。

## 【0012】

## 【課題を解決するための手段】

この目的の達成のため、本発明は、上位装置からのディスクアクセス要求に応じて、複数の物理ディスクに分散して格納された論理ボリューム上のアクセス要



求領域をアクセスする R A I D 制御装置において、各々異なる論理ボリュームを担当し、1つの論理ボリュームを複数の物理ディスクに分散して、冗長グループを組む複数の物理ディスク群と、前記論理ボリュームの論理フォーマット処理の進捗情報を管理するための管理テーブルを参照して、前記論理ボリュームの各領域の論理フォーマット要求を発行する複数の論理フォーマット管理モジュールと、前記論理フォーマット管理モジュールの処理要求に応じて、前記物理ディスクを実アクセスする複数の下位層モジュールとを有する。そして、前記各論理フォーマット管理モジュールは、前記複数の物理ディスク群を担当するため、前記各物理ディスク群の管理テーブルを有し、前記ディスクアクセス要求に応じて、前記管理テーブルを参照し、当該アクセス領域全てが、論理フォーマット済かを判定し、フォーマット済と判断した場合に、前記下位層モジュールにディスクアクセスを要求し、フォーマット済でなければ、論理フォーマット処理要求を発行し、前記ディスクアクセス要求を待ちキューにキューイングし、前記待ちキューにディスクアクセス要求がない時は、前記管理テーブルから未フォーマット領域を検索して、論理フォーマット処理要求を発行することを特徴とする。

## 【 0 0 1 3 】

本発明では、ホストアクセスのバックグラウンドで、物理ディスクの論理フォーマット処理を行うため、シーケンシャルに論理フォーマットを実行しながら、ホストアクセスを実行できる。

## 【 0 0 1 4 】

又、論理ボリューム上の各ブロックが未フォーマットかフォーマット済かの進捗情報をテーブルで管理するため、シーケンシャルフォーマット処理を実行しながら、ディスクアクセス要求を実行できるとともに、容易に、ディスクアクセス要求のあったアクセス領域がフォーマット済かを判定できるため、クイックフォーマットを実行しても、アクセス要求に対する応答を高速化できる。

## 【 0 0 1 5 】

しかも、冗長グループを構成する場合に、クイックフォーマットの進捗情報を冗長管理するため、クイックフォーマット処理も冗長化でき、一のユニットの障害時、他のユニットで、それまでの進捗情報を利用して、クイックフォーマット

処理を継続できる。

【 0 0 1 6 】

又、本発明では、好ましくは、一の前記論理フォーマット管理モジュールの異常に応じて、他の前記論理フォーマット管理モジュールで、前記一の論理フォーマット管理モジュールの担当する物理ディスク群の論理フォーマットを実行することにより、容易に、一のユニットの障害時、他のユニットで、それまでの進捗情報を利用して、クイックフォーマット処理を継続できる。

【 0 0 1 7 】

更に、本発明では、好ましくは、一の前記論理フォーマット管理モジュールの異常に応じて、他の前記論理フォーマット管理モジュールの管理テーブルから、前記一の論理フォーマット管理モジュールの担当する物理ディスク群の管理テーブルを、前記一論理フォーマット管理モジュール以外の他の論理フォーマット管理モジュールに復元する R A I D 管理モジュールを有する。これにより、一のユニットが故障しても、他のユニットでクイックフォーマットの冗長構成を構築できる。

【 0 0 1 8 】

更に、本発明では、好ましくは、前記 R A I D 管理モジュールは、R A I D 構成を記述した構成テーブルを参照し、前記復元を実行することにより、容易に冗長構成に復元できる。

【 0 0 1 9 】

更に、本発明では、好ましくは、前記管理テーブルは、前記論理ボリュームの論理フォーマット処理単位の進捗状況をビットマップで管理することにより、容易に 1 つのユニットで、複数の冗長グループの論理フォーマットの進捗状況を管理、更新できる。

【 0 0 2 0 】

【発明の実施の形態】

以下、本発明の実施の形態を、R A I D ストレージシステム、クイックフォーマット処理、クイックフォーマット処理の冗長構成、他の実施の形態の順で説明する。

## 【 0 0 2 1 】

## 〔 R A I D ストレージシステム 〕

図 1 は、本発明の一実施の形態の R A I D ストレージシステムの構成図であり、磁気ディスクを使用した R A I D (Redundant Arrays of Inexpensive Disk) システムを示す。図 1 に示すように、磁気ディスク制御装置（以下 F C U という） 1 は、計算機センタ（ホスト） 3 の大量のデータを R A I D ディスクドライブ（以下、ディスクという） 2 へ高速かつ、ランダムに読み書きが出来るシステムである。

## 【 0 0 2 2 】

F C U 1 は、C A (Channel Adapter) 1 0 / C M (Centralized Module) 1 2 / D A (Device Adapter) 1 6 の 3 つのファンクションモジュールによって構成されている。これらの 3 つのファンクションモジュールは、F C U 1 内部のアクションを行うための役割分担が決められており、処理の一極集中による負担を軽減している。また、モジュール間のコミュニケーション・インタフェースは、メッセージや A C B と呼ばれるものを用いて処理内容や、各モジュールの状態の伝達を行っている。

## 【 0 0 2 3 】

各ファンクション・モジュール 1 0、1 2、1 6 の役割分担の概要を以下に説明する。C A (Channel Adapter) 1 0 は、F C U 1 と H O S T 3 を結ぶホスト・インタフェースの制御をつかさどるモジュールである。C A 1 0 は、H O S T 3 A 1 0、C M 1 2、D A 1 6 を接続し、コマンド、データのやり取りを行う。この R A I D (Redundant Arrays of Inexpensive Disk) は、大型機向けの高価なディスクを、複数の小型機向けの小容量、安価なディスクで置き換え、データの保全、冗長性を保証するディスクシステムである。

## 【 0 0 2 4 】

図 2 は、図 1 の構成の詳細ブロック図であり、クイックフォーマットを実現するモジュールを示している。ここで、クイックフォーマットとは、実際の論理フォーマット化を行う前に、ホストアクセスを許可し、ホストアクセスのバックグラウンドで、論理フォーマット処理を実行するものを意味する。又、図 2 では、R

A I D 5 のシステムを示しており、4 つの冗長グループ A, B, C, D が設けられ、各グループは、4 台の物理ディスク 2-1~2-4、2-5~2-8、2-9~2-12、2-13~2-16 で構成される。

#### 【0025】

CM12 は、上位層モジュール 20 と、4 つのクイックフォーマットモジュール 21-1~21-4 と、RAID 管理モジュール 23 とを有する。DA16 は、下位層モジュール 22-1~22-8 を有する。上位層モジュール 20 は、論理フォーマット及びリード・ライト等を要求する。下位層モジュール 22-1~22-8 は、論理フォーマット及びリード・ライト等のディスクアクセスを管理する。

#### 【0026】

上位層モジュール 20 は、CA10 を介しホスト 3 からの要求を受けて、物理ディスクへの I/O が必要かを判定し、必要であれば、ディスクアクセス要求を下位層モジュール 22-1~22-8 に要求すること、及び、当モジュール内部でスケジュールされるライトバック等を下位層のモジュールに要求することを行う。RAID 管理モジュール 23 は、後述する図 18 以下で説明するように、各冗長グループに対する冗長構成の維持のための管理を行う。

#### 【0027】

クイックフォーマット管理モジュール 21-1~21-4 は、主に、ホストアクセスに関係なく、論理ボリューム上の各ブロックをシーケンシャルにフォーマット処理し、上位層のモジュール 20 からのディスクアクセス要求の指定ブロックによって、必要に応じた論理ボリューム上のブロックをフォーマット処理する機能を持つ。

#### 【0028】

ここでは、4 つの冗長グループ A, B, C, D に対し、4 つのクイックフォーマット管理モジュール 21-1~21-4 が設けられている。各管理モジュール 21-1~21-4 は、2 つの冗長グループを担当する。例えば、管理モジュール 21-1 は、冗長グループ A を担当し、更に、障害時には、冗長グループ D も担当する。図 2 の接続関係は、論理パスで示している。

## 【0029】

下位層のモジュール22-1～22-8は、上位層のモジュール20及びクイックフォーマット管理モジュール21-1～21-4からの要求を下位層キュー35に受けて、データの冗長性管理を含めた物理ディスク2-1～2-16へのI/O（入出力）動作を行う。ここでは、各冗長グループA, B, C, Dに対し、2つの下位層モジュール22-1～22-8を設け、二重化構成を採用している。

## 【0030】

各クイックフォーマット管理モジュール21-1～21-4は、フォーマット待ちキュー31と、過剰要求回避のためのフォーマット要求待ちキュー32と、冗長グループを構成する各論理ボリュームのフォーマット処理の進捗情報のテーブル33と、各論理ボリュームの各ブロックの論理フォーマット処理の進捗情報のテーブル34とを有する。

## 【0031】

次に、図3により、論理ボリュームと物理ディスクとの関係を説明する。全論理ボリュームを0～nとすると、図3に示すように、各冗長グループA, B, C, Dは、全論理ボリューム0～nの各々1部、例えば、1/4ずつの論理ボリューム0～1、1+1～P、…、s～nを担当する。各冗長グループは、RAID5の構成では、図16で説明するように、4つのディスクの内3つのディスクでデータを持ち、1つのディスクでチェックデータを持ち、冗長構成をとる。RAID1の構成では、各冗長グループが、2つのディスクで冗長構成を組み、各冗長グループが、全論理ボリューム0～nを担当する。

## 【0032】

論理ボリュームは、ホスト3のOSの認識単位であり、mブロックのデータで構成される。そして、RAID構成では、論理ボリュームの各ブロックが、複数（ここでは、4つ）の物理ディスク2-1～2-4に分散して配置される。例えば、冗長グループAの各論理ボリューム0～1のブロックは、冗長グループAを担当する4台の物理ディスク2-1～2-4に分散配置される。即ち、1つの論理ボリュームを複数の物理ディスクで担当する、所謂冗長構成をとる。

## 【0033】

図4は、図2の進捗情報テーブル33、34の説明図である。クイックフォーマット管理モジュール21-1～21-4は、クイックフォーマット中の各ブロックが未フォーマットかフォーマット済かの情報を持つ。ここで、情報の持ち方としては、論理ボリューム0～nごとに、ビットマップ34-0～34-nを持ち、未フォーマット/フォーマット済領域を区別する。

## 【0034】

また、各冗長グループでの論理ボリューム単位での進捗を管理する論理ボリューム管理テーブル33-Aを有し、各冗長グループの論理ボリューム単位のクイックフォーマット処理を管理する。これにより、論理ボリューム単位でのフォーマット処理が可能となる。また、フォーマット済みの論理ボリュームについてはビットマップの検索を行わないので、処理が遅れにくくなる。即ち、冗長グループA、B、C、Dに対して、担当する論理ボリュームの管理テーブル33-A、33-B、33-C、33-Dと、ビットマップ34-0～34-1、34-1+1～34-P、…、34-s～34-nを設ける。

## 【0035】

図5は、下位層モジュール22-1～22-8へ送出される要求に付随した要求内容テーブルR-Mの説明図である。

## 【0036】

テーブルR-Mは、要求の処理種別（リード、ライト、フォーマット）100と、処理領域（フォーマット領域、アクセス領域）101と、データ領域102とを有する。更に、リード/ライトアクセス時に使用されない領域に、フォーマットのための情報が設定される。

## 【0037】

即ち、フォーマットの種別（ワンポイントかシーケンシャルか）、冗長グループ内で処理上限数が超えたため、フォーマット待ちのフォーマット要求に対し、先に受け付けたフォーマット要求、後に受け付けたフォーマット要求104、フォーマット実施中のフォーマット要求に対し、先に受け付けたフォーマット要求、後に受け付けたフォーマット要求105、フォーマット実施中のフォーマット

要求と競合してフォーマット要求に対し、先に受け付けたフォーマット要求、後に受け付けたフォーマット要求 1 0 6 である。

【 0 0 3 8 】

これらのテーブル R - M は、要求時に、上位層モジュール 2 0、クイックフォーマット管理モジュール 2 1 - 1 ~ 2 1 - 4 が作成し、要求に付加する。

【 0 0 3 9 】

〔クイックフォーマット処理〕

次に、図 6 により、図 2 の上位層モジュール 2 0、クイックフォーマット管理モジュール 2 1 の処理を説明する。尚、管理モジュール 2 1 が、上位層のモジュールから要求があった場合の処理である。

【 0 0 4 0 】

( S 1 ) 上位層モジュール 2 0 は、ホストのアクセス対象がクイックフォーマットの対象の論理ボリュームかを判定する。アクセス要求は、論理ボリューム名、論理アドレスを含む。論理ボリューム管理テーブル 3 3 には、各論理ボリュームのクイックフォーマット済みを示すフラグが格納されている。上位層モジュール 2 0 は、この管理テーブル 3 3 を参照して、アクセス対象の論理ボリュームが、クイックフォーマットの対象かどうかを判定する。

【 0 0 4 1 】

( S 2 ) アクセス対象の論理ボリュームが、クイックフォーマットの対象外なら、フォーマットされているから、下位層モジュール 2 2 - 1 ~ 2 2 - 8 へ、通常のディスクアクセスを要求する。下位層モジュール 2 2 - 1 ~ 2 2 - 8 は、これを下位層キュー 3 5 にキューイングし、キューイング順にディスクをアクセスする。

【 0 0 4 2 】

( S 3 ) 逆に、クイックフォーマットが行われている論理ボリュームへホストからのアクセスがあった場合には、処理をクイックフォーマット管理モジュール 2 1 - 1 ~ 2 1 - 4 に渡す。クイックフォーマット管理モジュールは、アクセス領域が未フォーマットかどうかを対応する論理モジュールのビットマップ 3 4 - 0 ~ 3 4 - n で判定する。

【0043】

(S4) アクセス対象の領域が、フォーマット済みなら、フォーマットされているから、下位層モジュール22-1～22-8へ、通常のディスクアクセスを要求する。下位層モジュール22-1～22-8は、これを下位層キュー35にキューイングし、キューイング順にディスクをアクセスする。

【0044】

(S5) アクセス要求領域に未フォーマット領域があれば、その領域に対するフォーマット要求（ワンポイントフォーマット要求という）を発し、且つそのアクセスを、フォーマット待ちキュー31にエンキューする。

【0045】

次に、クイックフォーマット管理モジュール21-1～21-4の下位層モジュール22から論理フォーマット終了を受けた時の処理を、図7、図8及び図9で説明する。

【0046】

(S10) 論理フォーマット終了を受けると、先ず、進捗情報を更新する。即ち、図4の管理テーブル33、ビットマップ34-0～34-nを、終了した領域に応じて更新する。

【0047】

(S11) クイックフォーマットの対象とする全領域のフォーマットが完了したかを管理テーブル33、ビットマップ34-0～34-nを参照して、判定する。

【0048】

(S12) 全領域のフォーマットが完了したと判定すると、フォーマット待ちキュー31の検索条件を初期化し、フォーマット待ちキュー31を検索し、フォーマット待ちディスクアクセスがあれば、待ちキュー31から外し、下位層モジュール22へ通常アクセスを要求する。これを、待ちキュー31の全ディスクアクセスの検索が終了するまで繰り返し、全ディスクアクセスの検索が終了すると、処理を終了する。

【0049】



(S13) 全領域のフォーマットが完了していないと判定すると、フォーマット待ちキュー31の検索条件を初期化し、フォーマット待ちキュー31を検索し、フォーマット待ちディスクアクセスがあれば、そのアクセスの対象領域が、フォーマット済みかを、ビットマップ34-0~34-nを参照して、判定する。フォーマット済みである時は、待ちキュー31から外し、下位層モジュール22へ通常アクセスを要求する。フォーマット済みでないときは、待ちキュー31に残す。これを、待ちキュー31の全ディスクアクセスの検索が終了するまで繰り返し、全ディスクアクセスの検索が終了すると、この処理を終了する。

#### 【0050】

(S14) 次に、論理フォーマット要求処理を行う。この処理は、基本的には、図7に示すように、待ちキュー31にアクセス要求が残っているときは、そのアクセスの対象領域の論理フォーマット要求（ワンポイントフォーマットという）を下位層モジュール22に要求する。一方、待ちキュー31にアクセス要求が残っていない時は、内部スケジュールにより決めた領域の論理フォーマット（シーケンシャルフォーマットという）を下位層モジュール22に要求する。この論理フォーマット要求処理は、図8及び図9で詳細に後述する。

#### 【0051】

このように、アクセスがない時は、シーケンシャルに論理ボリュームの各領域を論理フォーマット処理し、アクセスがあると、アクセス対象領域が未フォーマットかを判定し、未フォーマットであるときは、そのアクセス対象領域を論理フォーマット処理する。その後、そのフォーマットされた領域にアクセスを行う。

#### 【0052】

このため、ホストアクセスの待ち時間を低減し、シーケンシャルに論理フォーマット処理することができる。更に、図8及び図9の論理フォーマット要求処理では、これらの論理フォーマット要求の多重化を行う。

#### 【0053】

(S20) 論理フォーマットが終了した要求処理の要求内容テーブルR-M（図5参照）を調べ、終了した要求処理にリンクしている後に受け付けたフォーマット要求があるかを判定する。リンクしている処理がないと、図9のように、こ

の処理を終了する。

【0054】

(S21) リンクしている処理がある時は、先ず、冗長グループの処理数制限を行うための待ちキュー32に要求処理があるかを調べる。キューイングされた処理がある時は、次の要求処理を、冗長グループの処理数制限を行うための待ちキュー32の先頭に設定する。一方、キューイングされた処理がない場合には、次の処理にシーケンシャルフォーマット処理に設定する。

【0055】

(S22) 次に、次の要求がシーケンシャルフォーマットかを判定する。シーケンシャルフォーマットである時は、内部スケジュールによりフォーマット領域を決定する。この時、前述のビットマップ34-0～34-nの進捗情報を参照して、フォーマット領域を決定できる。

【0056】

(S23) 論理フォーマット要求の対象領域に未フォーマット領域があるかを判定する。この時、前述のビットマップ34-0～34-nの進捗情報を参照して、フォーマット領域を決定できる。未フォーマット領域がない場合には、論理フォーマットの必要がないため、この要求を下位層モジュール22に発行することなく、処理を終了する。

【0057】

(S24) 論理フォーマット要求の対象領域に未フォーマット領域がある場合には、冗長グループに対するフォーマット要求数が最大値を超えるかを判定する。即ち、管理モジュール21は、下位層モジュール22に発行し、下位層モジュール22で未処理のフォーマット要求数をカウントしている。このカウント値を参照し、判定する。冗長グループに対するフォーマット要求数が最大値を超えたと判定した時は、このフォーマット要求処理を、冗長グループの処理数制限のための待ちキュー32にキューイングし、処理を終了する。

【0058】

(S25) 冗長グループに対するフォーマット要求数が最大値を超えない場合には、このフォーマット要求の対象領域が、下位層モジュール22のキュー35

にキューイングされたフォーマット要求の領域と競合するかを、前述の要求内容テーブル R-M (図 5 参照) を参照して、判定する。下位層モジュール 22 のキュー 35 にキューイングされたフォーマット要求の領域と競合する場合には、この競合する両フォーマット要求の要求内容テーブル R-T の競合関係欄 106 に先後関係を設定し、終了する。一方、下位層モジュール 22 のキュー 35 にキューイングされたフォーマット要求の領域と競合しない場合には、フォーマット要求を下位層モジュール 22 に発行し、終了する。

## 【0059】

図 10 乃至図 14 は、クイックフォーマット処理の動作説明図である。図 10 に示すように、シーケンシャルフォーマット S を実行中に、上位層モジュール 20 からディスクアクセス要求があり、ワンポイントフォーマット要求 1～8 が順で発行された例で説明する。

## 【0060】

次に実行されるシーケンシャルフォーマット要求が、該当する論理ボリュームの実施フォーマットキュー（下位層キュー）35-0 に入っているとすると、論理ボリューム 0 の下位層キュー 35-0 に、ワンポイントフォーマット要求 1、2、6 が順でキューイングされる。一方、ワンポイントフォーマット要求 5 は、ワンポイントフォーマット要求 1 と競合するため、ワンポイントフォーマット要求 1 にリンクする。

## 【0061】

同様に、論理ボリューム 1 の下位層キュー 35-1 に、ワンポイントフォーマット要求 3、4 が順でキューイングされる。一方、ワンポイントフォーマット要求 7 は、ワンポイントフォーマット要求 3、4 と競合するが、初めに、要求 3 と競合するため、ワンポイントフォーマット要求 3 にリンクする。

## 【0062】

ワンポイントフォーマット要求 8 は、ワンポイントフォーマット要求 7 と競合するが、実施中フォーマットキュー内の競合を意識すれば良いため、ワンポイントフォーマット要求 4 にリンクする。更に、下位層キュー 35-0、35-1 に対するフォーマット要求数が最大値を越える場合には、冗長グループ内の待ち

キュー 32 に入る。

【0063】

この時、フォーマット待ちキュー 31 には、ワンポイントフォーマット要求 1、2、3、4、5、6、7、8 の基となるアクセス要求（ライト要求）Wr 1～Wr 8 が、キューイングされ、要求内容テーブル R-T（図 5 参照）のリンク情報の設定で、図 10 のように、リンクする。

【0064】

次に、図 11 に示すように、シーケンシャルフォーマット S，ワンポイントフォーマット 1、3 が完了すると、論理ボリューム 0 の下位層キュー 35-0 では、実行されたシーケンシャルフォーマット要求 S とワンポイントフォーマット要求 1 が追い出され、ワンポイントフォーマット要求 2 が先頭となり、最後尾の要求 6 の次に、フォーマット待ちキュー 31 のアクセス要求（ライト要求）Wr 1 がキューイングされる。一方、ワンポイントフォーマット要求 5 は、競合したワンポイントフォーマット要求 1 が実行されたため、リンクを解かれ、アクセス要求に移る。

【0065】

同様に、論理ボリューム 1 の下位層キュー 35-1 では、ワンポイントフォーマット要求 3 が実行されたため、ワンポイントフォーマット要求 4 が先頭となり、冗長グループ待ちキュー 32 の待たされた要求 n が移動し、フォーマット待ちキュー 31 のアクセス要求 Wr 3 の順でキューイングされる。一方、ワンポイントフォーマット要求 7 は、ワンポイントフォーマット要求 3 が完了したが、要求 4 と競合するため、今度は、ワンポイントフォーマット要求 8 にリンクする。

【0066】

このように、アクセス要求の対象領域が未フォーマット領域を含む場合には、ワンポイントフォーマット要求が発行され、このフォーマット要求は、フォーマット実施確認待ちキュー（下位層キュー）へエンキューされ、下位層へフォーマット処理依頼される。このためフォーマット要求が複数、下位層へ要求される事になる。

【0067】

一方、シーケンシャルフォーマットは、所定の領域（例えば、1ブロック）単位での処理が終わる毎に、上記キューの最後尾にフォーマット処理としてキューイングされる。このとき、図12に示すように、シーケンシャルフォーマット処理では、ビットマップテーブル34の先頭から未フォーマット領域について検索し、フォーマット要求を行う。

## 【0068】

又、このクイックフォーマットにおいて、ワンポイントフォーマット処理を複数行う事が可能となったため、フォーマット領域が重なる事がある。この場合、対象領域について先のワンポイントフォーマット処理にリンクし、ワンポイントフォーマット処理が終わるのを待って、対象領域の未フォーマット領域をフォーマットする。

## 【0069】

フォーマット領域重複のために待ちとなっているワンポイントフォーマットのフォーマット領域について、全領域がフォーマット済みであれば、フォーマット処理は実行せず、リンクを外す。

## 【0070】

シーケンシャルフォーマットの行われていない論理ボリュームに対するワンポイントフォーマット要求があった時についても、ワンポイントフォーマット処理はフォーマット実施確認待ちキューへのエンキューの後、下位層に要求され、これら複数の要求処理は同時並行的に行われる。

## 【0071】

更に、フォーマット処理要求については処理要求数が多くなりすぎると、前述のように、ワンポイントフォーマット完了後に、そのアクセス要求がキューイングされるため、アクセス要求の実行が遅くなる。これを防ぐ目的から、図12に示すように、冗長グループ毎、論理グループ毎のフォーマット処理数についての制限を設け、この制限によりフォーマット処理を一時保留する。

## 【0072】

又、図13に示すように、ビットマップ34-0～34-nを用いて、1ビット当たり数ブロックの未フォーマット／フォーマット済の情報を進捗管理し、こ

の進捗管理において1ビットのフォーマット処理が終わる毎に、進捗情報は更新される。これにより、より細かい進捗情報の管理が可能となる。

## 【0073】

また、クイックフォーマット実施中のホストアクセスによるワンポイントフォーマット処理とシーケンシャルフォーマット処理との競合時、実施中シーケンシャルフォーマットからの復帰が早くなり、ホストアクセスへの応答が遅れにくくなる。さらに、フォーマット実施時のビットマップの確認が1ビットのみで済むため、処理が速い。

## 【0074】

上述の進捗情報更新方法では、更に、フォーマット処理単位での進捗情報の更新を行う。この更新方法では、進捗情報の更新を、フォーマット処理終了を契機にして行う為、ワンポイントフォーマット中に、複数回更新を行わずに済み、フォーマット終了後に一度更新するだけなので、結果的にホストアクセスへの応答が速くなる。

## 【0075】

また、フォーマット処理単位での進捗情報更新を行う上で、進捗情報の更新はテーブルのコピーという方法で効率的に行う事ができる。又、図14に示すように、複数のフォーマット処理完了毎に、複数ビットを一度に更新することもできる。

## 【0076】

図15乃至図17は、論理フォーマットの説明図である。論理フォーマットは、ホストの物理ディスクへのデータアクセスが行えるように、物理ディスク2を論理的にフォーマットするものであり、基本的には、1トラックのデータ領域200を、ホストの処理単位（例えば、512バイトのブロック）に区切り、各々のブロック200-1～200-pに、ID（識別子）と、CRC等のチェックコードを書き込むものである。

## 【0077】

又、図16のRAID-5では、3ブロックのデータを、1ブロックずつ物理ディスク2-1、2-2、2-3に割り当て、そのパリティデータを、物理ディ

スク 2-4 に割り当てる。この時、論理フォーマット処理では、これらのブロックに関連をもたせるため、ID にリンク番号を付す。例えば、物理ディスク 2-1、2-2、2-3、2-4 の対応するブロックの ID を、ID 1, ID 2, ID 3, ID P と付す。

## 【 0 0 7 8 】

同様に、図 1 7 に示すように、RAID-1 のミラーリングでは、冗長データブロックに、同一の ID 番号を付す。例えば、ミラーリングを構成する物理ディスク 2-1、2-2 の冗長データブロックに、ID-1 を付す。

## 【 0 0 7 9 】

このように、論理ボリューム上の各ブロックが未フォーマットかフォーマット済かの進捗情報をビットマップ 3 4 で管理するため、ディスクアクセス要求があった場合には、当該アクセス領域全てが、フォーマット済かを判定する。そして、フォーマット済となっていれば、下位層モジュールにディスクアクセスを要求し、フォーマット済でなければ、ディスクアクセスに先んじて、論理フォーマット処理要求を発行し、ディスクアクセス要求を待ちキューにキューイングする。

## 【 0 0 8 0 】

一方、待ちキューにディスクアクセス要求がない時は、ビットマップ 3 4 から未フォーマット領域を検索して、論理フォーマット処理要求を発行する。更に、論理フォーマット処理要求の実行終了に応じて、進捗情報を管理するビットマップ 3 4 を更新する。

## 【 0 0 8 1 】

このため、シーケンシャルフォーマット処理を実行しながら、ディスクアクセス要求を実行できるとともに、容易に、ディスクアクセス要求のあったアクセス領域がフォーマット済かを判定できるため、クイックフォーマットを実行しても、アクセス要求に対する応答を高速化できる。しかも、シーケンシャルフォーマットへの復帰も、ビットマップを参照するため、容易となる。

## 【 0 0 8 2 】

又、未フォーマット領域に対する複数のホストアクセスの論理フォーマット要求同士、ホストアクセスの論理フォーマット要求とシーケンシャルフォーマット

要求との競合を判定し、一方を待たせ、リンク処理するため、論理フォーマット要求を連続的に発行しても、これらを整理して、論理フォーマット要求を処理できる。このため、ホストアクセスへの応答が高速にできる。

## 【0083】

更に、論理フォーマット要求を連続的に発行しても、冗長グループ、論理ボリューム単位に、下位層のキュー数を制限し、制限数を越えたフォーマット処理要求を一時保留するため、冗長グループ、論理ボリューム単位のフォーマット処理の遅れを防止できる。

## 【0084】

しかも、論理ボリュームに対し、管理テーブル33、ビットマップ34を持ち、論理ボリューム単位且つブロック単位にフォーマット済かを管理するため、論理フォーマット単位でのフォーマット処理が可能となる。従って、フォーマット済の論理ボリュームは、ビットマップ34の検索を行わないため、処理の遅れを防止できる。進捗情報の管理をビットマップ34で行うため、参照処理、更新処理を高速化でき、結果的に、ディスクアクセス要求を高速にできる。

## 【0085】

## [クイックフォーマット処理の冗長構成]

次に、冗長グループ構成におけるクイックフォーマット処理の冗長構成を説明する。図18、図19、図20は、図2の冗長グループ構成におけるクイックフォーマット処理の冗長構成の説明図である。

## 【0086】

図2の構成では、4つの冗長グループA、B、C、Dに対し、4つのクイックフォーマット管理モジュール21-1～21-4が設けられている。各管理モジュール21-1～21-4は、2つの冗長グループを担当する。即ち、図19の論理接続図に示すように、管理モジュール21-1は、冗長グループAを担当し、更に、障害時には、冗長グループDも担当する。同様に、管理モジュール21-2、21-3、21-4は、各々冗長グループB、C、Dを担当し、更に、障害時には、冗長グループA、B、Cも担当する。図19の接続関係は、論理パスで示している。RAID管理モジュール23のRAID構成テーブル23は、図



20に示すように、設定され、2重化制御する。

【0087】

この場合に、図18及び図19に示すように、管理モジュール21-1に、冗長グループAの進捗情報テーブルA（2つの待ちキュー31、32、管理テーブル33、ビットマップ34）の他に、冗長グループDの進捗情報テーブルのコピーD'（少なくとも、管理テーブル33、ビットマップ34であり、この例では、2つの待ちキュー31、32を含む）を設ける。

【0088】

同様に、管理モジュール21-2、21-3、21-4に、各々冗長グループB、C、Dの進捗情報テーブルB、C、Dの他に、障害時担当する冗長グループA、B、Cの進捗情報テーブルA'、B'、C'を設ける。

【0089】

即ち、各々クイックフォーマット管理モジュールをもつ複数ユニット1~4（21-1~21-4）が、それぞれ進捗情報を持つという形で、進捗情報の冗長化をとる。即ち、図18、図20に示すように、進捗情報を持つ各ユニットは、サイクリックにペアが組まれて、二重化している。これに合わせて、論理フォーマットの進捗情報を二重化する。

【0090】

これにより、二重化された一方の進捗情報が消失しても、進捗情報を持つ1のユニットが、ハードウェアの故障等により、故障ハード上のビットマップテーブルを含む進捗情報を失っても、他方から復元し、又は他方を用いて、クイックフォーマットを継続できる。

【0091】

図21、図22、図23、図24は、図2の冗長グループ構成におけるクイックフォーマット処理の冗長制御の説明図である。図18の冗長構成において、進捗情報を持つ1のユニットが、ハードウェアの故障等により、故障ハード上のビットマップテーブルを含む進捗情報を失い、冗長性の無い状態となったとき、図21、図22に示すように、他に冗長を組めるユニットがあれば、残ったユニットから再度ビットマップや進捗情報が復元され、スペアユニットと冗長構成をと

ることができる。

【0092】

図21、図22で説明すると、進捗情報を持つ1のユニット1（クイックフォーマット管理モジュール21-1）が、ハードウェアの故障等により、故障ハード上のビットマップテーブルを含む進捗情報を失い、冗長性の無い状態となったとする。

【0093】

この場合、図21、図22に示すように、他に冗長を組めるユニット2，3，4（クイックフォーマット管理モジュール21-2，21-3，21-4）があるため、残ったユニット2，3，4の進捗情報テーブルから再度ビットマップや進捗情報を復元し、スペアユニット2，3に、ユニット1の冗長グループA，Dの進捗情報テーブルD‘，A’を追加する。これにより、進捗情報テーブルの冗長構成をとることができる。

【0094】

このような冗長制御は、RAID管理モジュール23が行う。図24は、ユニット障害時のRAID管理モジュール23が実行する冗長制御フロー図である。

【0095】

（S30）RAID管理モジュール23は、クイックフォーマットの進捗情報テーブルを保有するユニット（図21、図22では、クイックフォーマット管理モジュール21-1）から故障の通知を受ける。

【0096】

（S31）RAID管理モジュール23は、図20の構成テーブル36を参照して、再度冗長を組むことができる他のユニット（クイックフォーマット管理モジュール）があるかを判定する。再度冗長を組むことができる他のユニットがない場合には、処理を終了する。

【0097】

（S32）再度冗長を組むことができる他のユニットが有る場合には、図20の構成テーブル36から、図18の時計回りのサイクリックのペアの規則に従い、図21、図22に示すように、ユニット2，3に、ユニット1のクイックフォ

ーマット進捗管理テーブルを構築する。そして、処理を終了する。この結果、構成テーブル 3 6 は、図 2 3 に示すようになる。

【 0 0 9 8 】

また、同様に、故障したユニット 1 を正常ユニットと交換し、正常状態となると、再び初期の冗長構成になるように復元する事ができる。即ち、図 2 1 の状態で、ユニット 1 を正常ユニットと交換した場合には、図 2 4 の処理と逆の処理により、図 1 8 の冗長構成に自動復帰する。

【 0 0 9 9 】

〔他の実施の形態〕

前述の実施の形態では、図 2 のような冗長構成の R A I D で説明したが、これ以外の冗長構成のストレージシステムに適用できる。又、物理ディスクは、磁気ディスク、光ディスク、光磁気ディスク、各種のストレージデバイスを適用できる。

【 0 1 0 0 】

（付記 1）上位装置からのディスクアクセス要求に応じて、複数の物理ディスクに分散して格納された論理ボリューム上のアクセス要求領域をアクセスする R A I D 制御装置において、各々異なる論理ボリュームを担当し、1 つの論理ボリュームを複数の物理ディスクに分散して、冗長グループを組む複数の物理ディスク群と、前記論理ボリュームの論理フォーマット処理の進捗情報を管理するための管理テーブルを参照して、前記論理ボリュームの各領域の論理フォーマット要求を発行する複数の論理フォーマット管理モジュールと、前記論理フォーマット管理モジュールの処理要求に応じて、前記物理ディスクを実アクセスする複数の下位層モジュールとを有し、前記各論理フォーマット管理モジュールは、前記複数の物理ディスク群を担当するため、前記各物理ディスク群の管理テーブルを有し、前記ディスクアクセス要求に応じて、前記管理テーブルを参照し、当該アクセス領域全てが、論理フォーマット済かを判定し、フォーマット済と判断した場合に、前記下位層モジュールにディスクアクセスを要求し、フォーマット済でなければ、論理フォーマット処理要求を発行し、前記ディスクアクセス要求を待ちキューにキューイングし、前記待ちキューにディスクアクセス要求がない時は、

前記管理テーブルから未フォーマット領域を検索して、論理フォーマット処理要求を発行することを特徴とする R A I D 制御装置。

【 0 1 0 1 】

(付記 2) 一の前記論理フォーマット管理モジュールの異常に応じて、他の前記論理フォーマット管理モジュールで、前記一の論理フォーマット管理モジュールの担当する物理ディスク群の論理フォーマットを実行することを特徴とする付記 1 の R A I D 制御装置。

【 0 1 0 2 】

(付記 3) 一の前記論理フォーマット管理モジュールの異常に応じて、他の前記論理フォーマット管理モジュールの管理テーブルから、前記一の論理フォーマット管理モジュールの担当する物理ディスク群の管理テーブルを、前記一論理フォーマット管理モジュール以外の他の論理フォーマット管理モジュールに復元する R A I D 管理モジュールを有することを特徴とする付記 1 の R A I D 制御装置。

【 0 1 0 3 】

(付記 4) 前記 R A I D 管理モジュールは、R A I D 構成を記述した構成テーブルを参照し、前記復元を実行することを特徴とする付記 3 の R A I D 制御装置。

【 0 1 0 4 】

(付記 5) 前記管理テーブルは、前記論理ボリュームの論理フォーマット処理単位の進捗状況をビットマップで管理することを特徴とする付記 1 の R A I D 制御装置。

【 0 1 0 5 】

(付記 6) 前記クイックフォーマット管理モジュールは、前記下位層モジュールからの論理フォーマット処理要求の実行終了に応じて、前記進捗情報を管理する管理テーブルを更新し、且つ前記論理フォーマット処理要求の実行終了に応じて、前記待ちキューを検索し、前記論理フォーマットの終了した前記ディスクアクセス要求を前記下位層モジュールに発行することを特徴とする付記 1 の R A I D 制御装置。

## 【 0 1 0 6 】

（付記 7）上位装置からのディスクアクセス要求に応じて、各々異なる論理ボリュームを担当し、1つの論理ボリュームを複数の物理ディスクに分散して、冗長グループを組む複数の物理ディスク群のアクセス要求領域をアクセスする R A I D 制御方法において、前記論理ボリュームの論理フォーマット処理の進捗情報を管理するための管理テーブルを参照して、前記論理ボリュームの各領域の論理フォーマット要求を発行する論理フォーマット要求ステップと、前記ディスクアクセス要求に応じて、前記管理テーブルを参照し、当該アクセス領域全てが、論理フォーマット済かを判定するステップと、フォーマット済と判断した場合に、前記物理ディスクを実アクセスする下位層モジュールにディスクアクセスを要求するステップと、フォーマット済でなければ、前記論理フォーマット処理要求を発行し、前記ディスクアクセス要求を待ちキューにキューイングするステップと、前記下位層モジュールからの前記論理フォーマット処理の終了通知に応じて、冗長性を持つ前記管理テーブルを更新するステップとを有することを特徴とする R A I D 制御方法。

## 【 0 1 0 7 】

（付記 8）一の論理フォーマット管理モジュールの異常に応じて、他の論理フォーマット管理モジュールで、前記一の論理フォーマット管理モジュールの担当する物理ディスク群の論理フォーマットを実行するステップを有することを特徴とする付記 7 の R A I D 制御方法。

## 【 0 1 0 8 】

（付記 9）一の論理フォーマット管理モジュールの異常に応じて、他の論理フォーマット管理モジュールの管理テーブルから、前記一の論理フォーマット管理モジュールの担当する物理ディスク群の管理テーブルを、前記一論理フォーマット管理モジュール以外の他の論理フォーマット管理モジュールに復元するステップを更に有することを特徴とする付記 7 の R A I D 制御方法。

## 【 0 1 0 9 】

（付記 1 0）前記復元ステップは、R A I D 構成を記述した構成テーブルを参照し、前記復元を実行するステップからなることを特徴とする付記 9 の R A I D

制御方法。

【0110】

（付記11）前記管理テーブルは、前記論理ボリュームの論理フォーマット処理単位の進捗状況をビットマップで管理することを特徴とする付記7のRAID制御方法。

【0111】

（付記12）前記論理フォーマット処理要求の実行終了に応じて、前記待ちキューを検索し、前記論理フォーマットの終了した前記ディスクアクセス要求を前記下位層モジュールに発行するステップを更に有することを特徴とする付記7のRAID制御方法。

【0112】

【発明の効果】

本発明では、ホストアクセスのバックグラウンドで、物理ディスクの論理フォーマット処理を行うため、シーケンシャルに論理フォーマットを実行しながら、ホストアクセスを実行できる。

【0113】

又、論理ボリューム上の各ブロックが未フォーマットかフォーマット済かの進捗情報をテーブルで管理するため、シーケンシャルフォーマット処理を実行しながら、ディスクアクセス要求を実行できるとともに、容易に、ディスクアクセス要求のあったアクセス領域がフォーマット済かを判定できるため、クイックフォーマットを実行しても、アクセス要求に対する応答を高速化できる。

【0114】

しかも、冗長グループを構成する場合に、クイックフォーマットの進捗情報を冗長管理するため、クイックフォーマット処理も冗長化でき、一のユニットの障害時、他のユニットで、それまでの進捗情報を利用して、クイックフォーマット処理を継続できる。

【図面の簡単な説明】

【図1】

本発明の一実施の形態のストレージシステムの構成図である。

【図 2】

本発明の一実施の形態の冗長構成のストレージシステムの構成図である。

【図 3】

図 2 の冗長グループ、論理ボリューム、物理ディスクの関係図である。

【図 4】

図 2 の管理テーブル、ビットマップの構成図である。

【図 5】

図 2 の下位層モジュールへの処理要求に付随する要求内容テーブルの説明図である。

【図 6】

図 2 の上位層モジュール、クイックフォーマット管理モジュールの上位層の要求処理フロー図である。

【図 7】

図 2 のクイックフォーマット管理モジュールの下位層モジュールからのフォーマット終了に応答する処理フロー図である。

【図 8】

図 7 のフォーマット要求処理フロー図（その 1）である。

【図 9】

図 7 のフォーマット要求処理フロー図（その 2）である。

【図 10】

図 8 乃至図 9 のフォーマット要求処理の動作説明図（その 1）である。

【図 11】

図 8 乃至図 9 のフォーマット要求処理の動作説明図（その 2）である。

【図 12】

図 8 乃至図 9 のフォーマット要求処理の動作説明図（その 3）である。

【図 13】

図 4 のビットマップの更新動作説明図である。

【図 14】

図 4 のビットマップの他の更新動作説明図である。

【図 1 5】

図 2 の論理フォーマットの説明図である。

【図 1 6】

図 2 の R A I D 5 における論理フォーマットの説明図である。

【図 1 7】

図 2 の R A I D 1 における論理フォーマット説明図である。

【図 1 8】

本発明の一実施の形態の冗長グループにおけるクイックフォーマット進捗管理テーブルの冗長構成の説明図である。

【図 1 9】

図 1 8 の本発明の一実施の形態の冗長グループにおけるクイックフォーマットの冗長構成の説明図である。

【図 2 0】

図 1 8 の本発明の一実施の形態の冗長グループにおけるクイックフォーマット進捗管理テーブルの冗長構成のための構成テーブルの説明図である。

【図 2 1】

本発明の一実施の形態のユニット故障時の冗長グループにおけるクイックフォーマット進捗管理テーブルの再冗長構成の説明図である。

【図 2 2】

図 2 1 の本発明の一実施の形態の冗長グループにおけるクイックフォーマットの再冗長構成の説明図である。

【図 2 3】

図 2 1 の本発明の一実施の形態の冗長グループにおけるクイックフォーマット進捗管理テーブルの最冗長構成のための処理フロー図である。

【図 2 4】

図 2 3 の処理による構成テーブルの説明図である。

【符号の説明】

- 1 ストレージ制御装置
- 2 物理ディスク



3 ホストコンピュータ

10 チャンネルアダプター

12 中央モジュール

16 デバイスアダプター

20 上位層モジュール

21-1～21-4 クイックフォーマット管理モジュール

22-1～22-8 下位層モジュール

23 R A I D 管理モジュール

31 フォーマット待ちキュー

32 冗長グループ待ちキュー

33 論理フォーマット管理テーブル

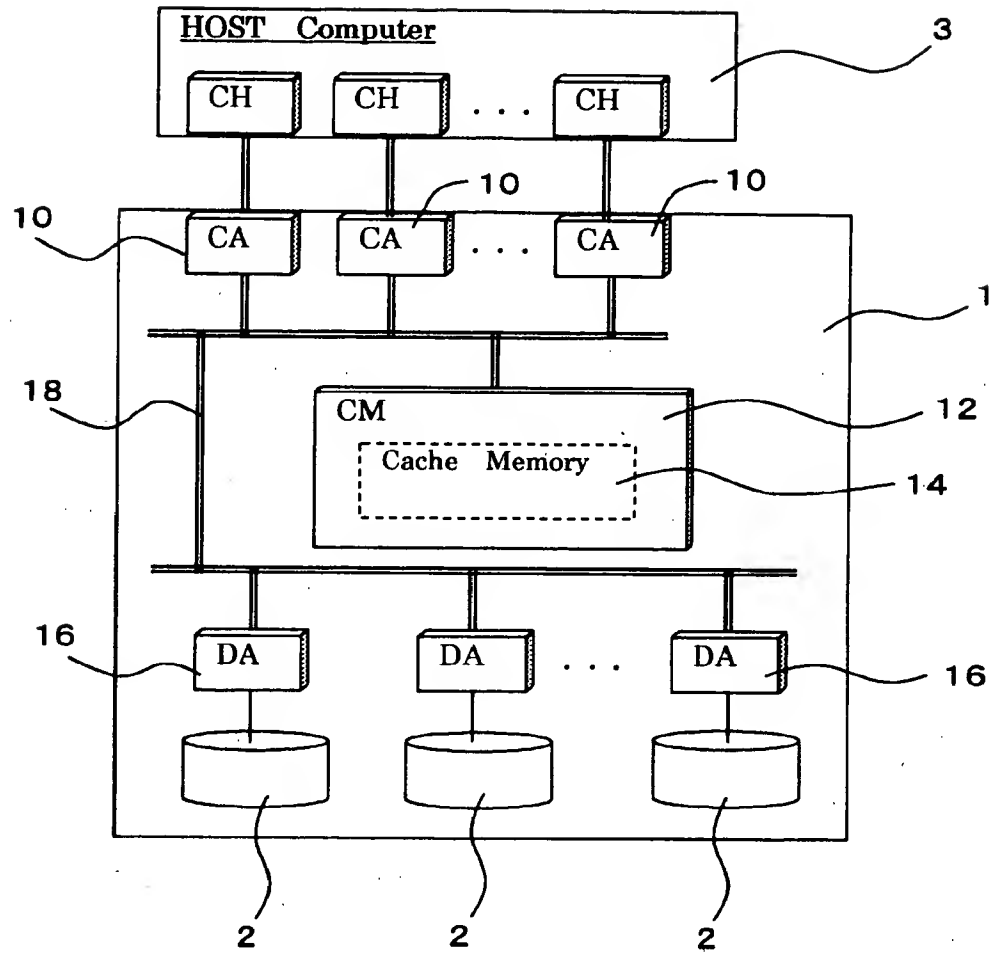
34 論理モジュールの進捗情報ビットマップ

35 下位層キュー

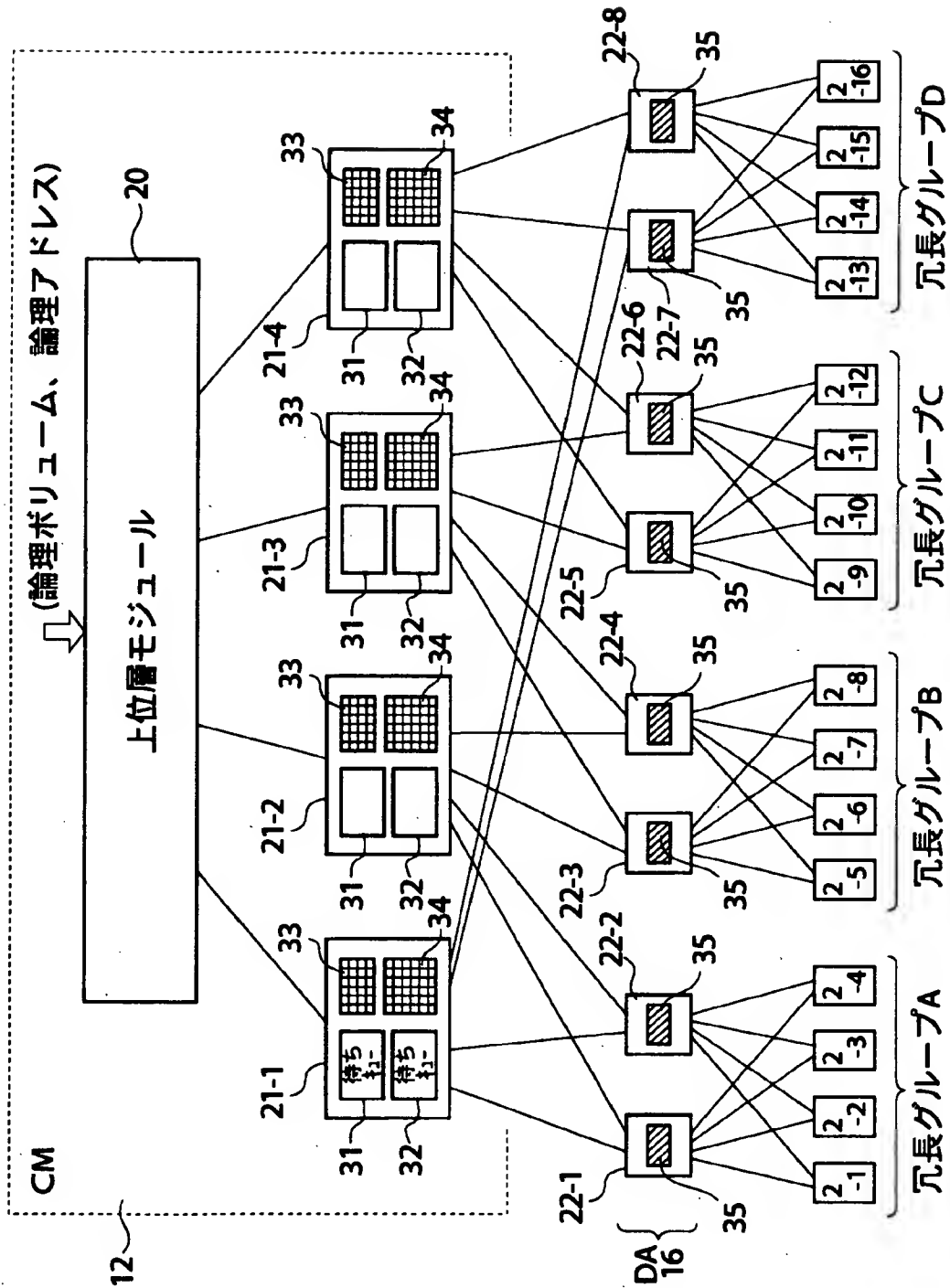
36 R A I D 構成テーブル

【書類名】 図面

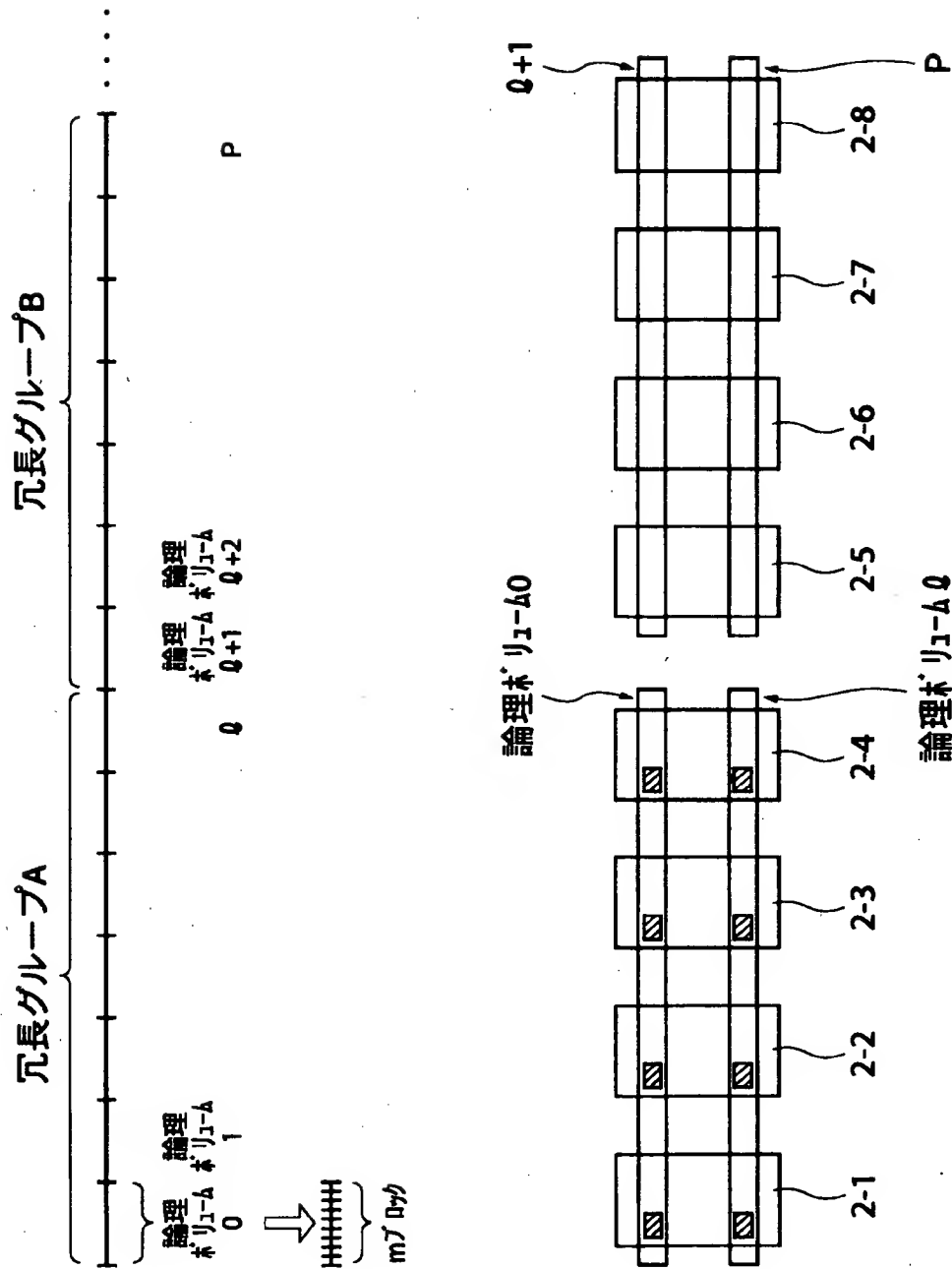
【図 1】



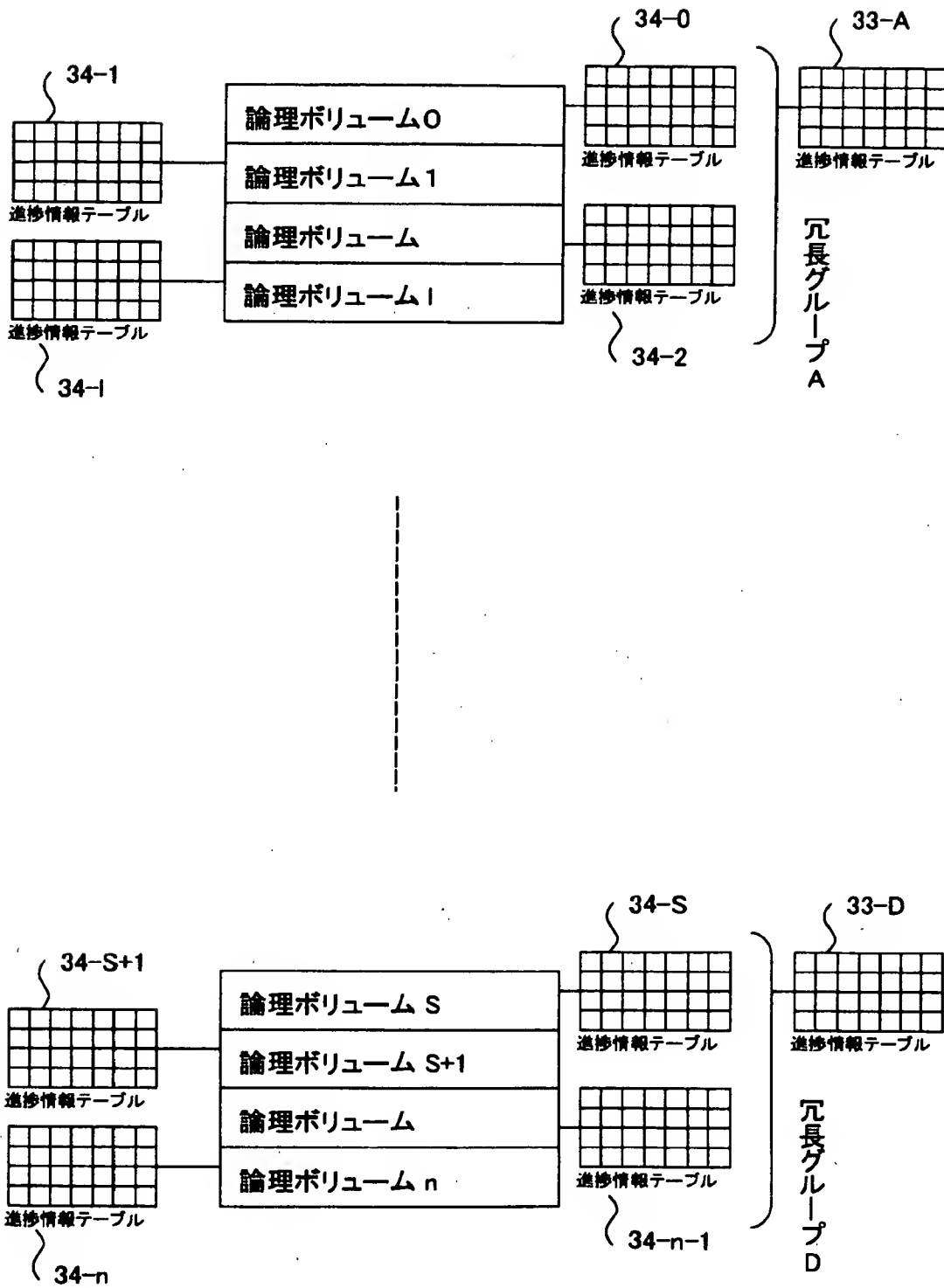
【図 2】



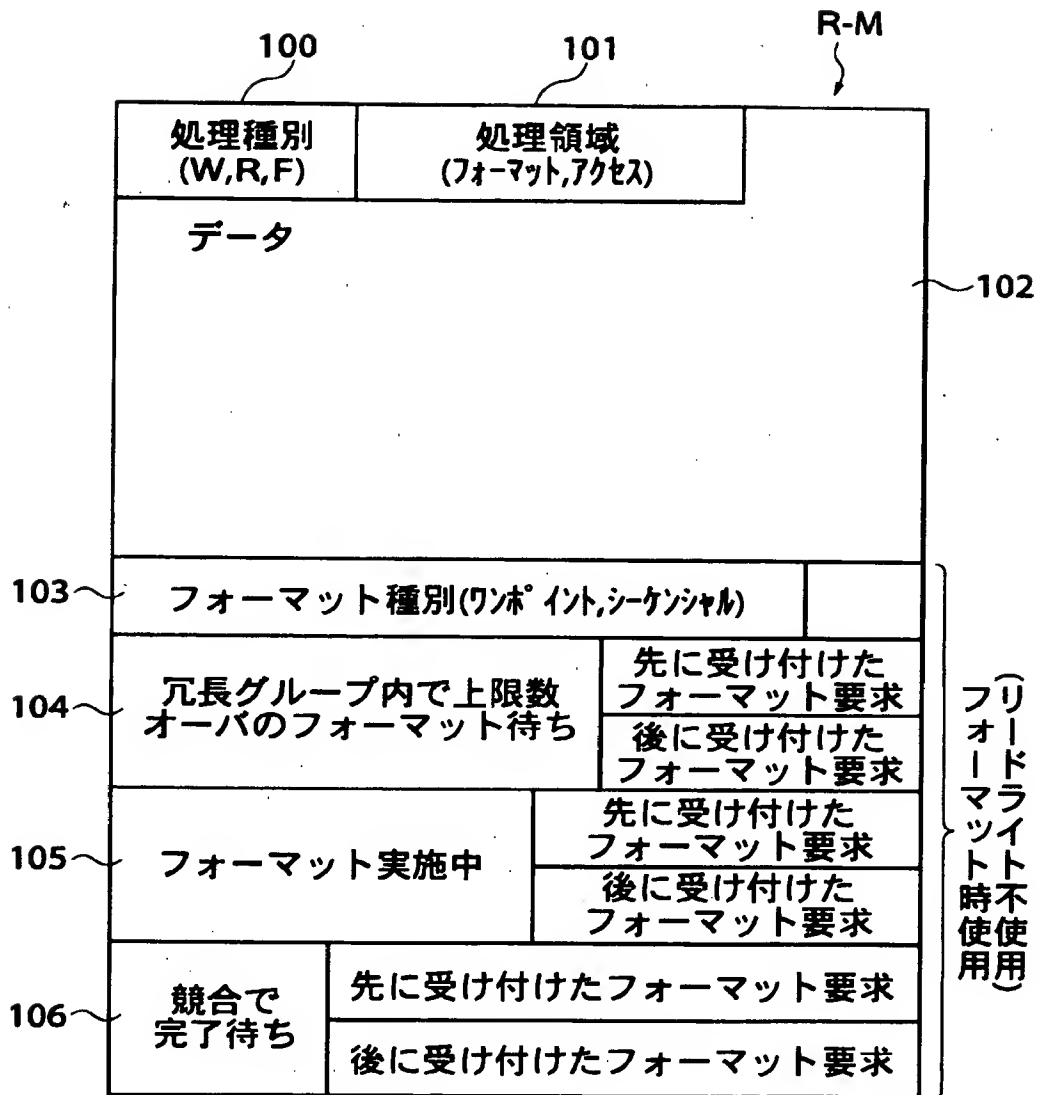
【図 3】



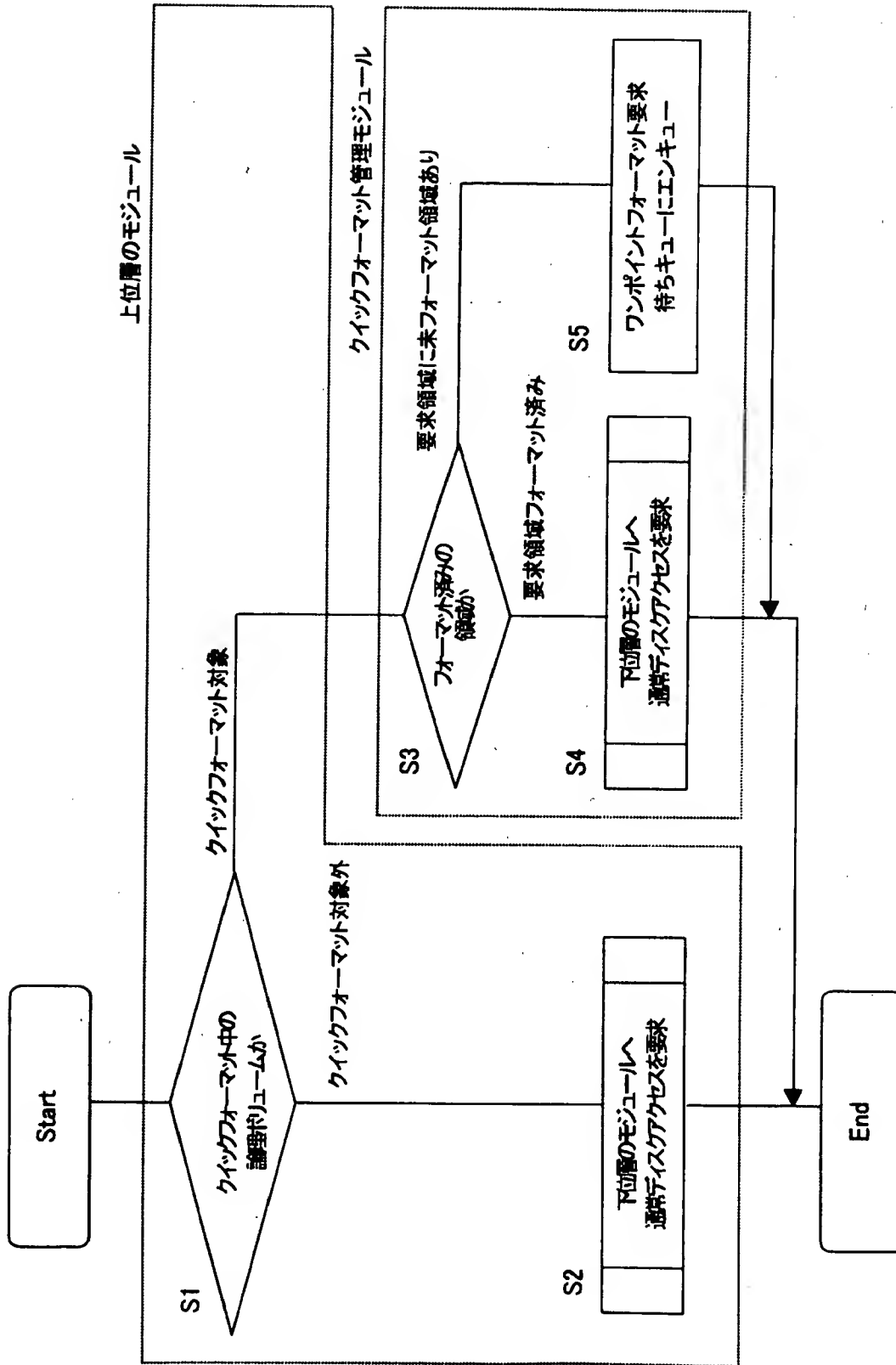
【図 4】



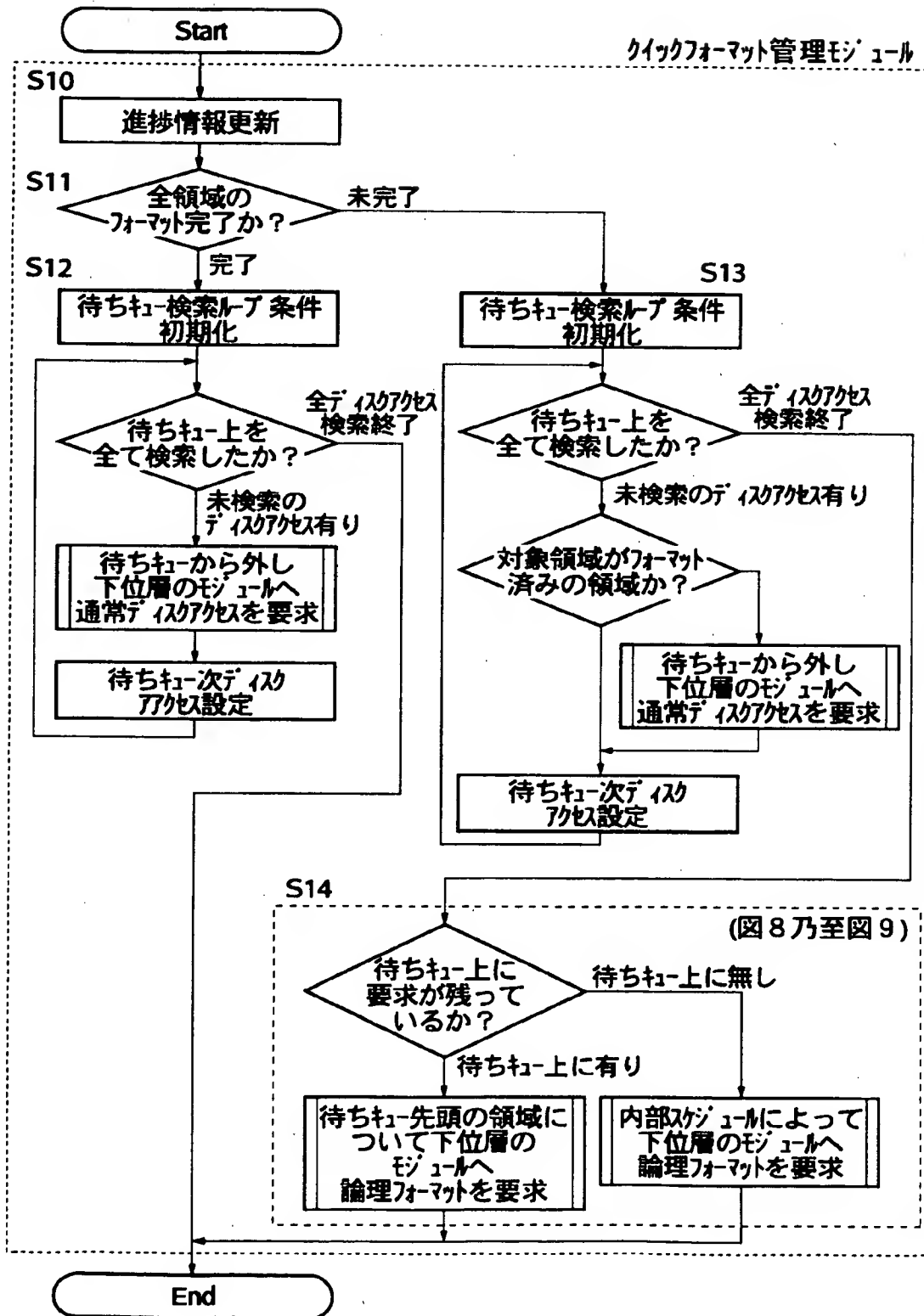
【図 5】



【図 6】

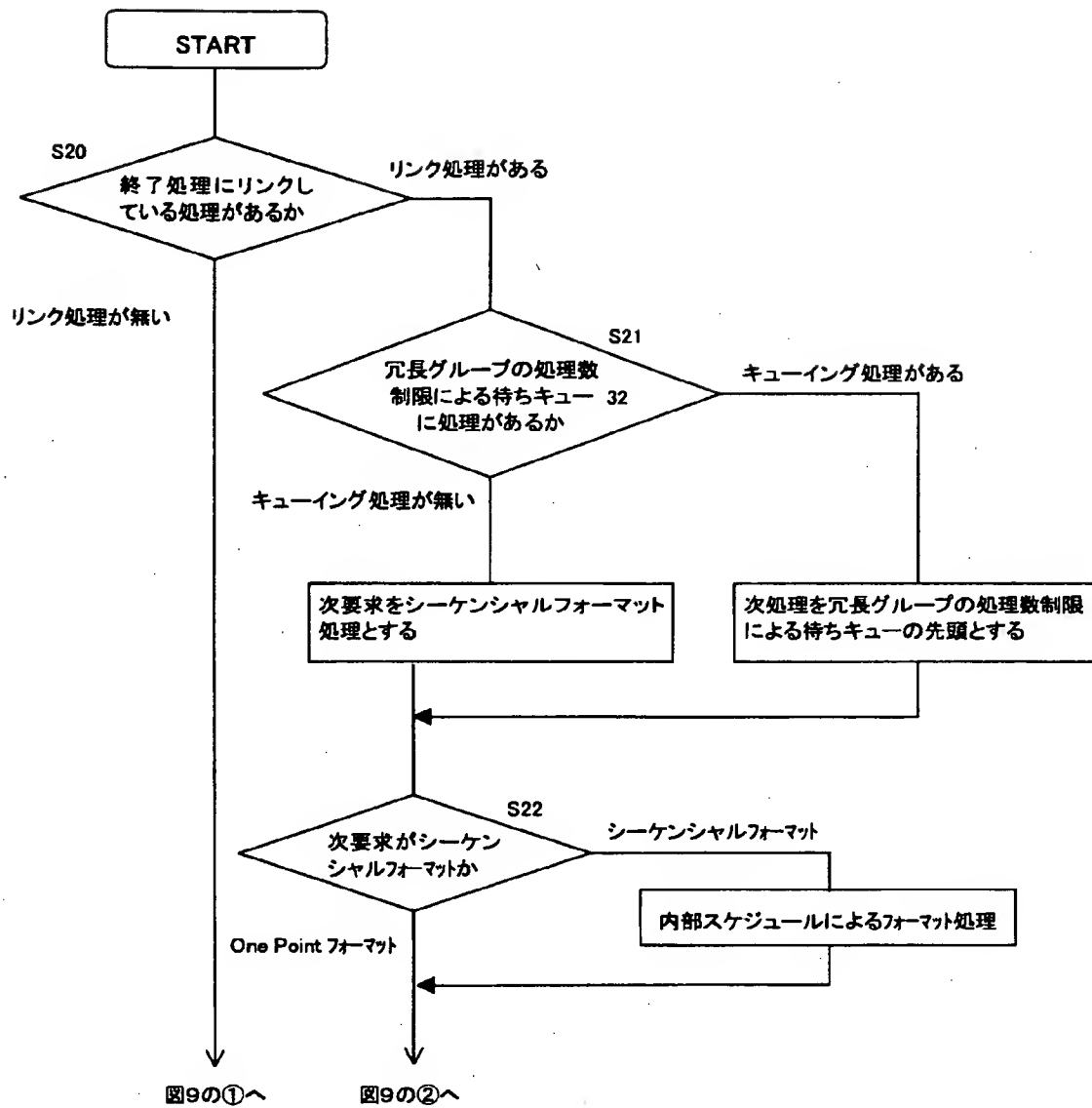


【図 7】

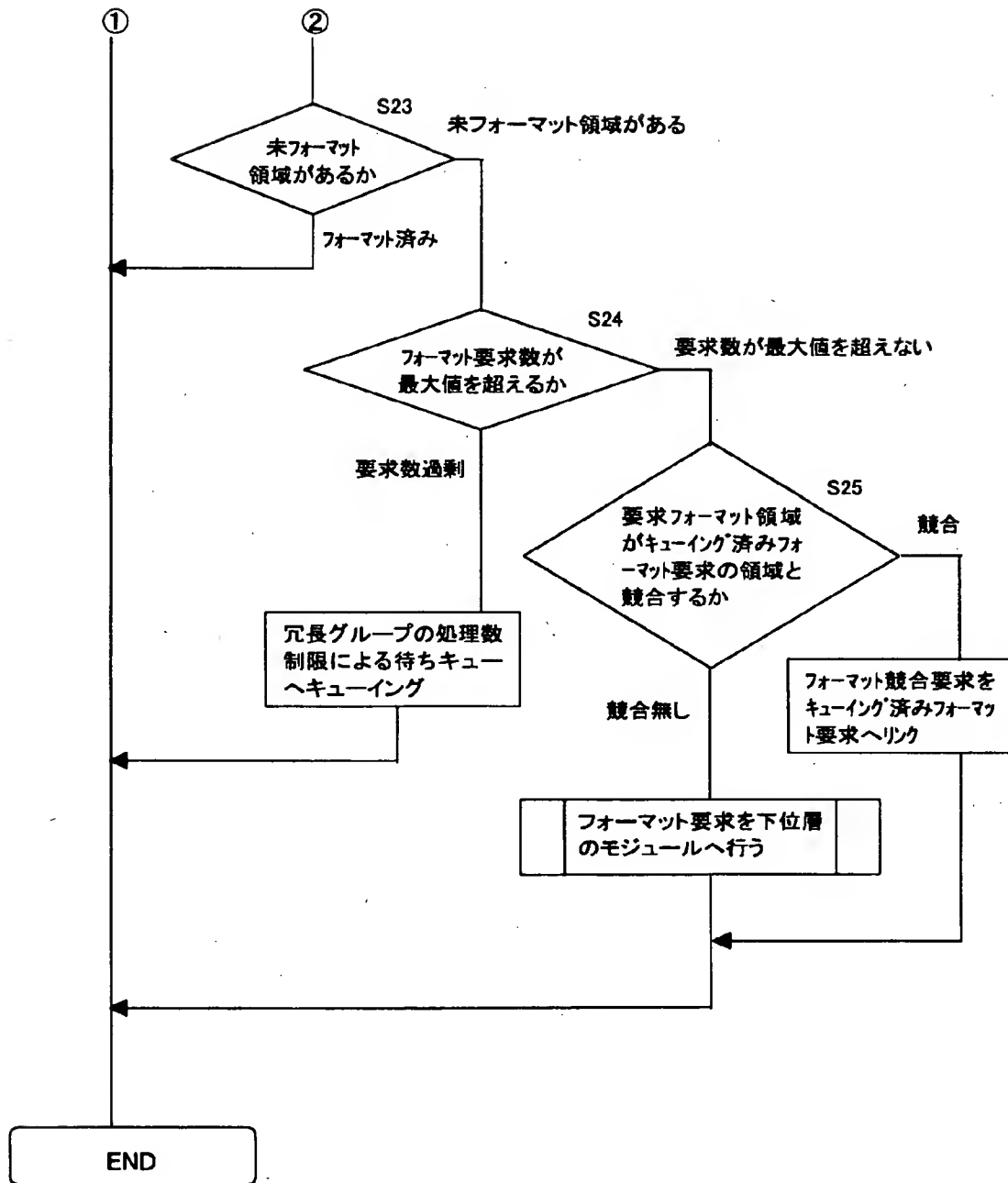




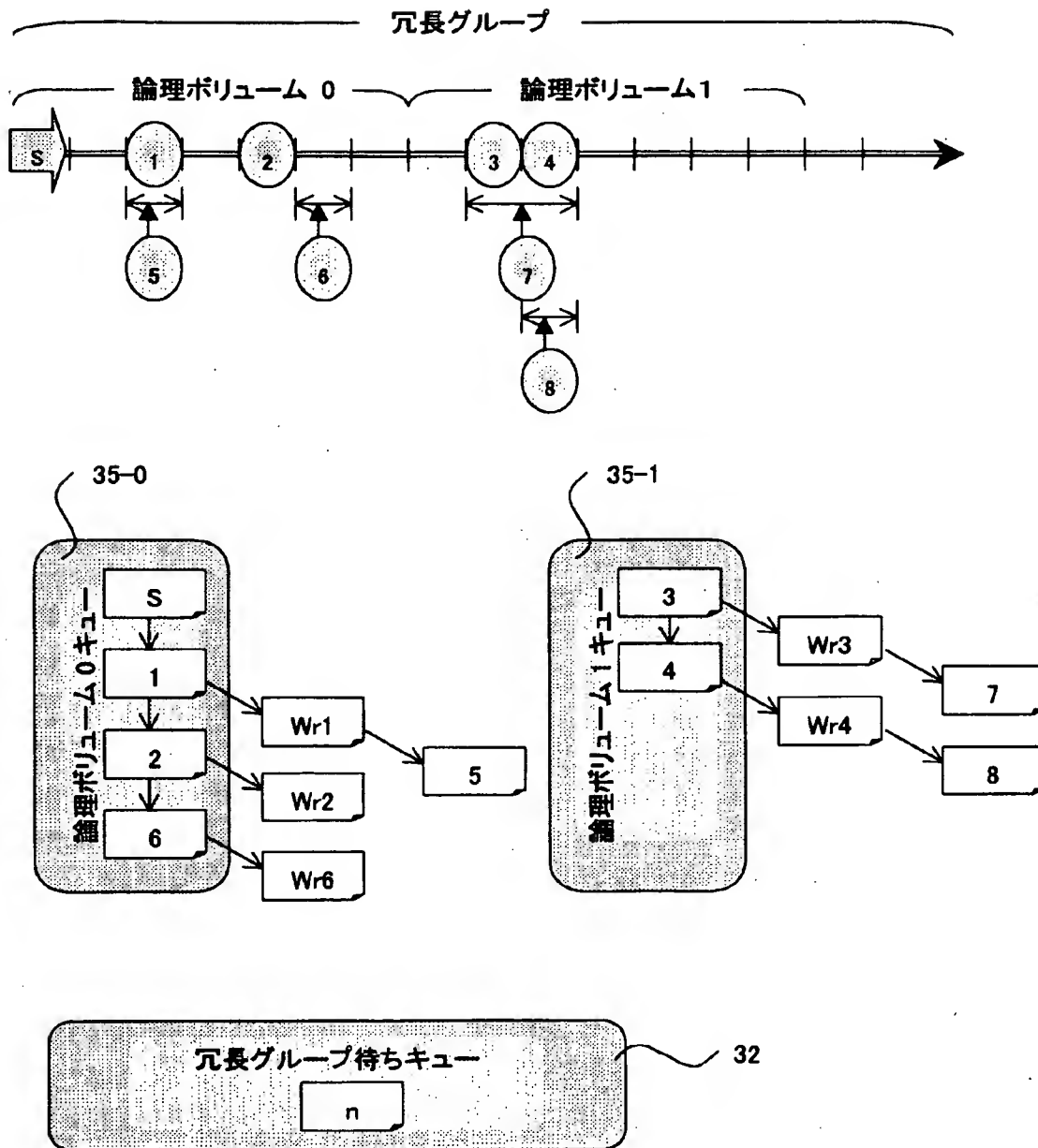
【図 8】



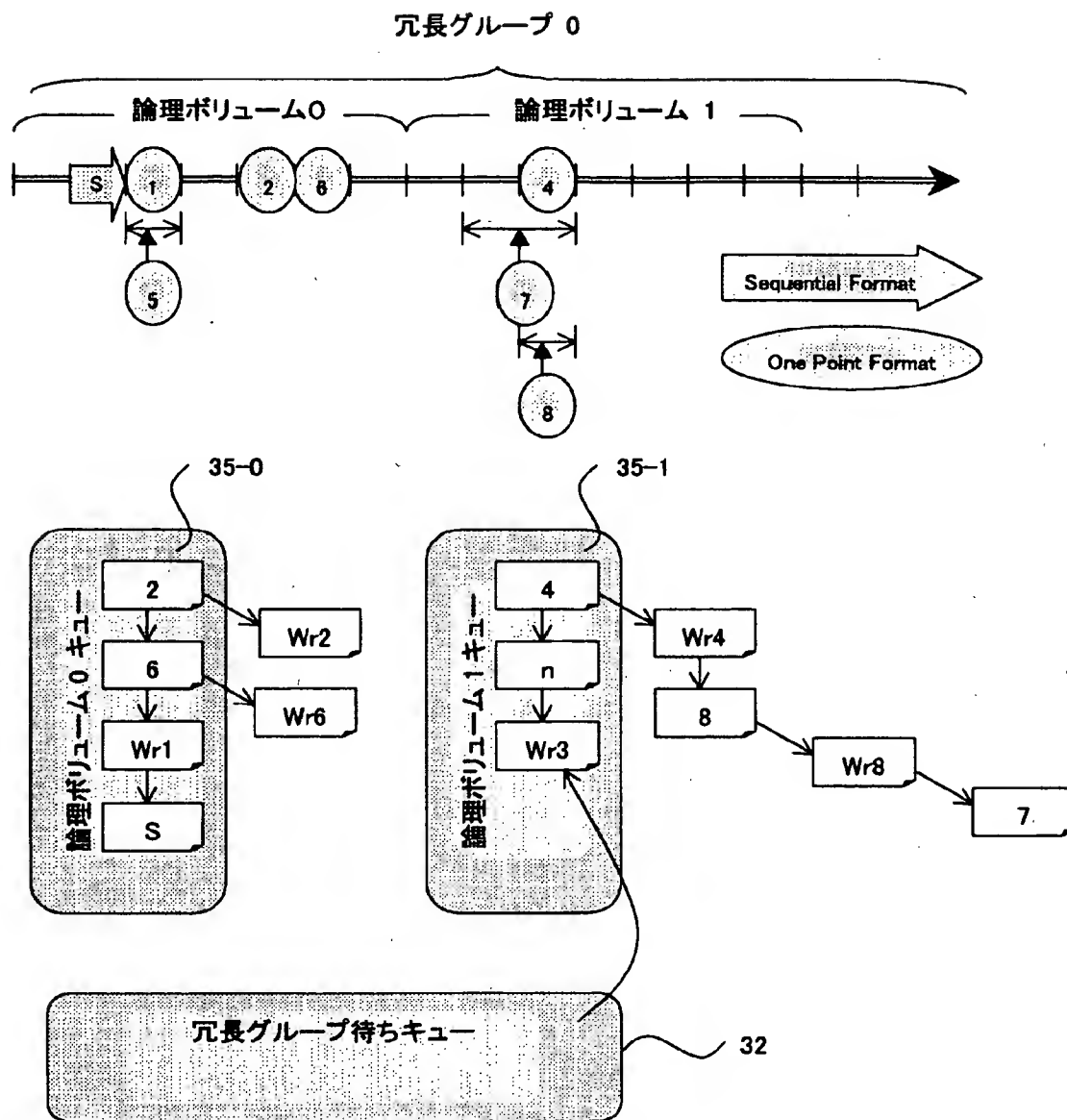
【図 9】



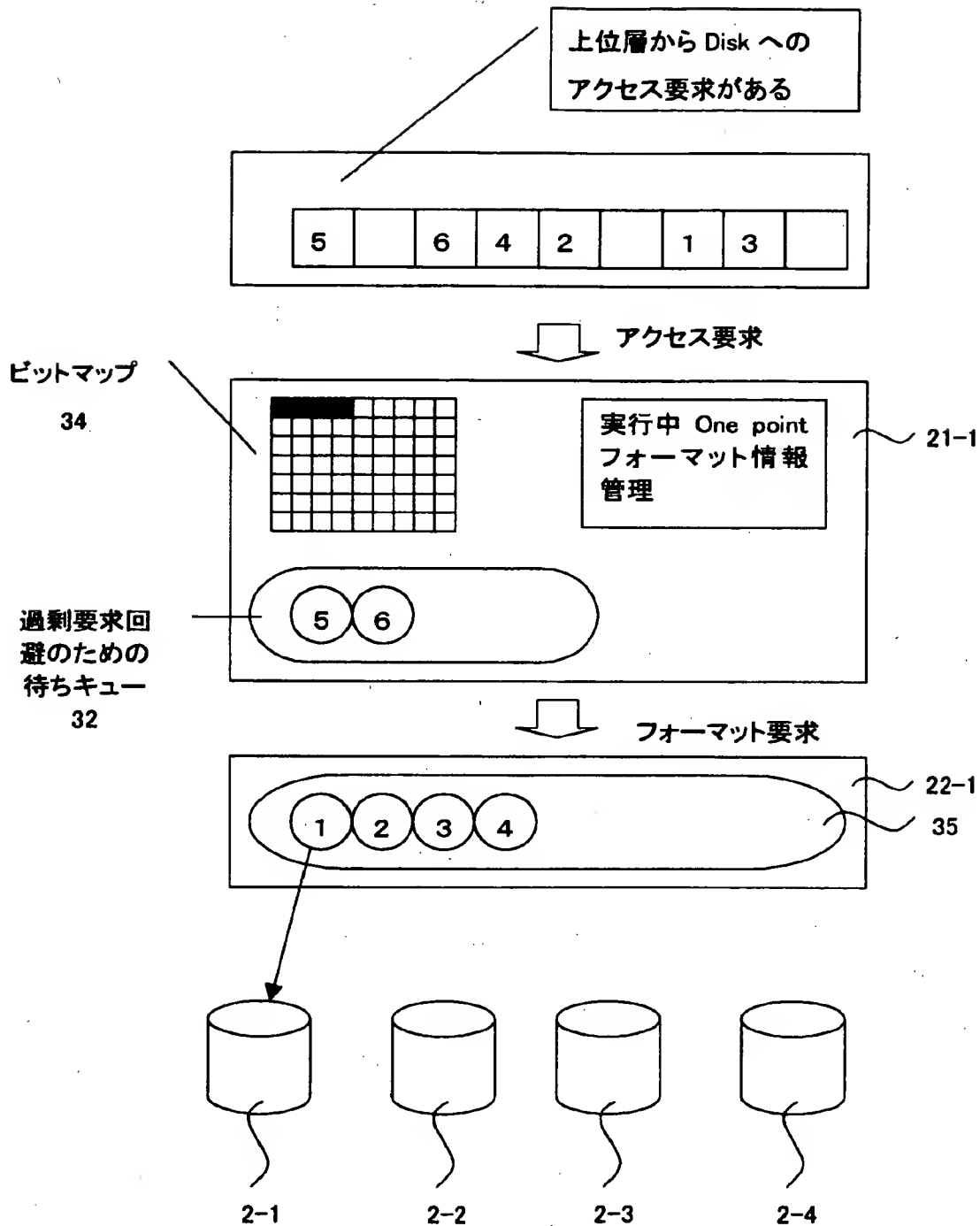
【図10】



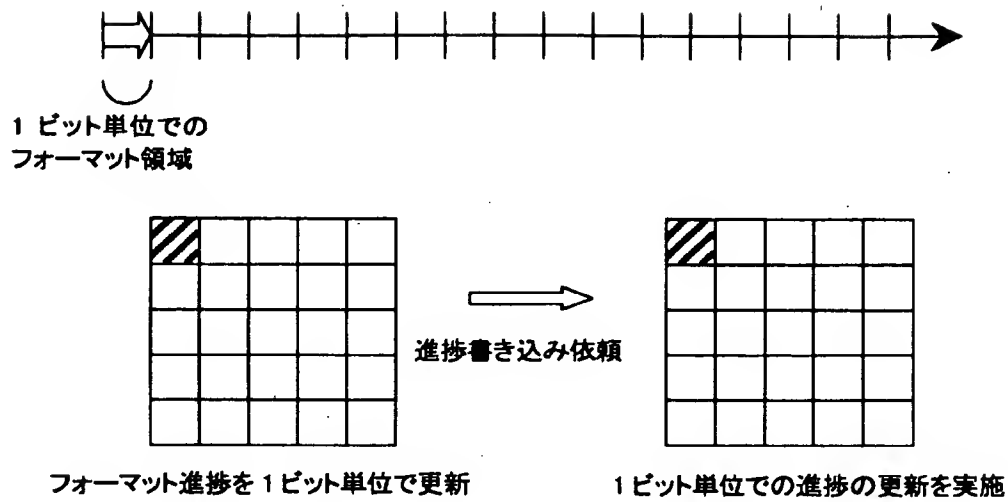
【図 11】



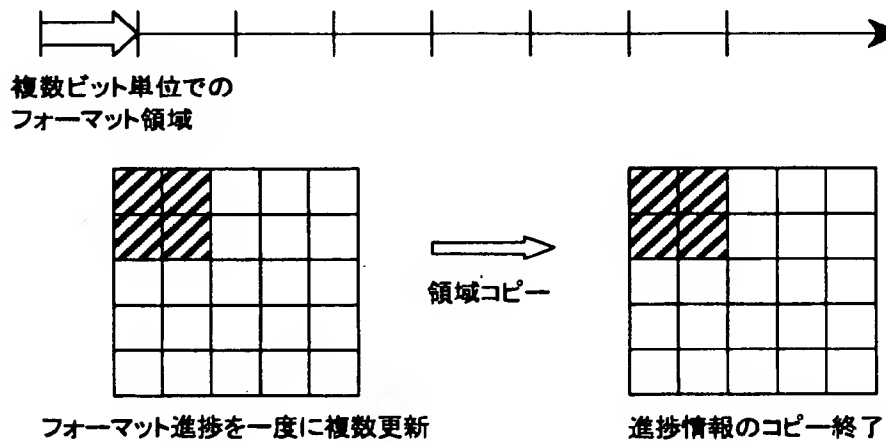
【図 12】



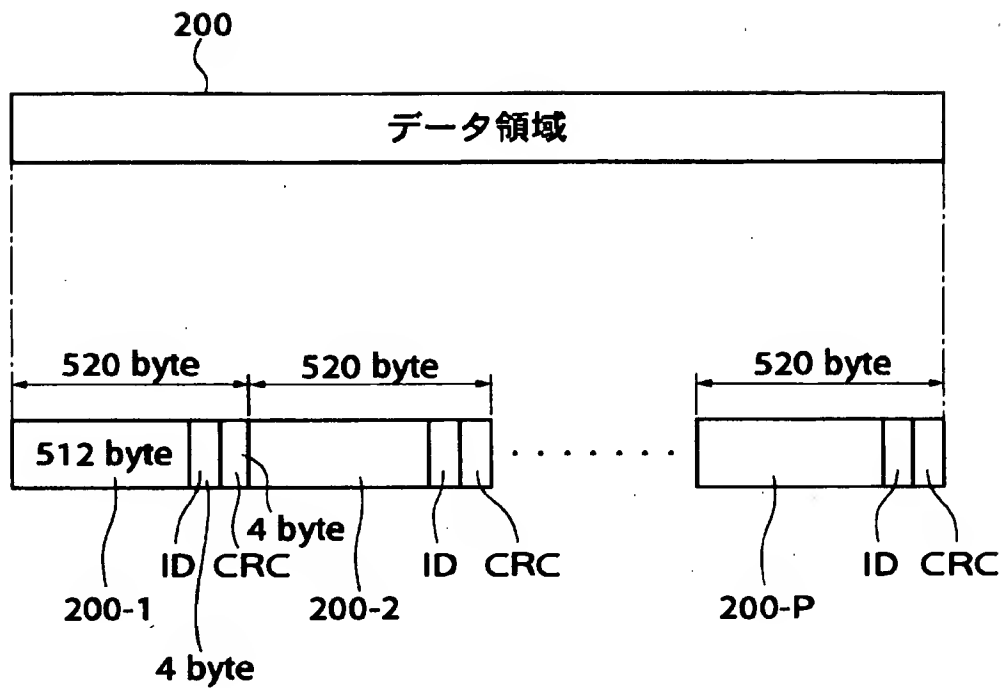
【図 13】



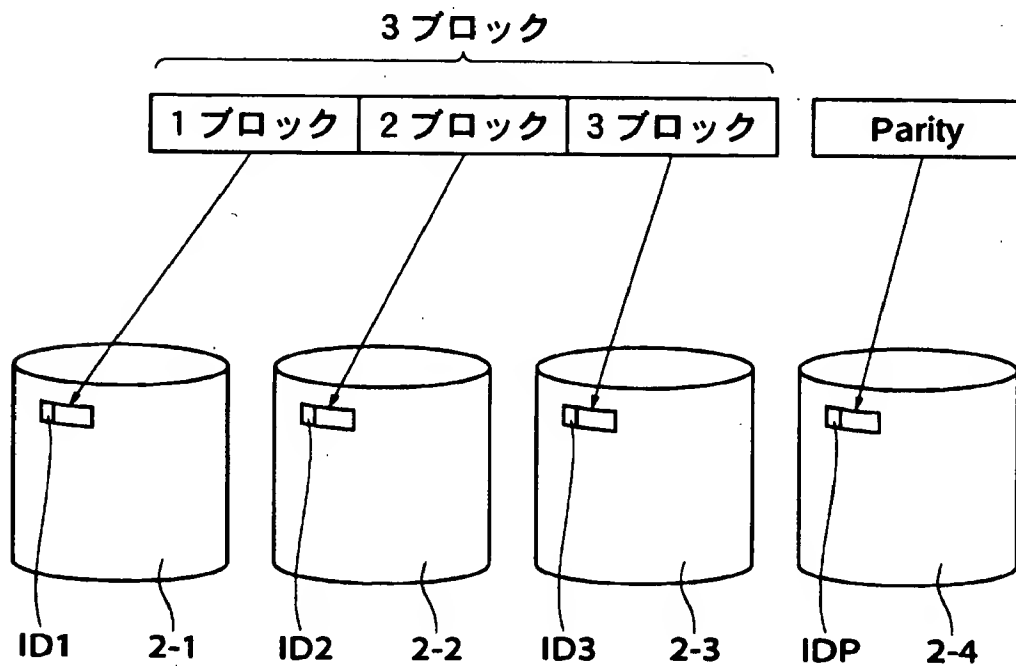
【図 14】



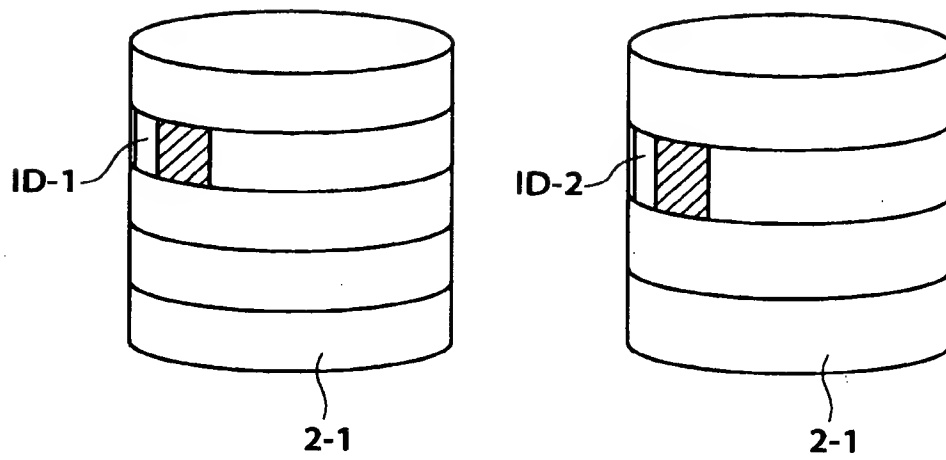
【図 1 5】



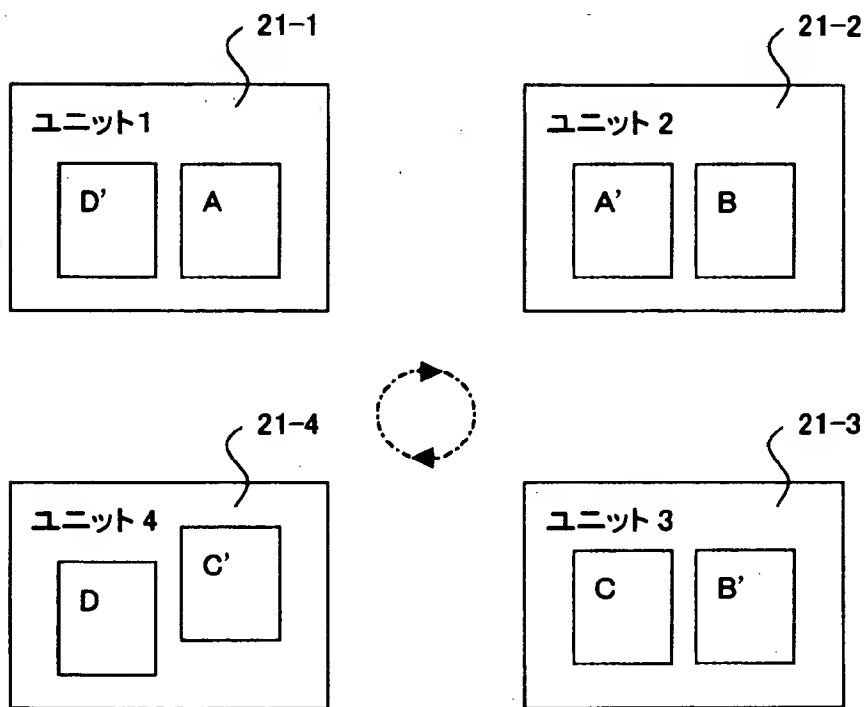
【図 1 6】



【図 17】

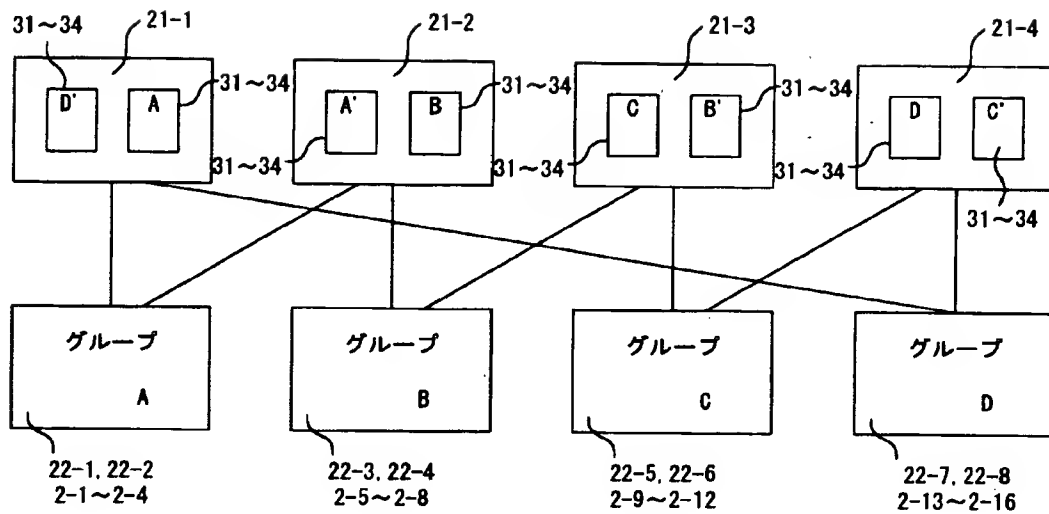


【図 18】





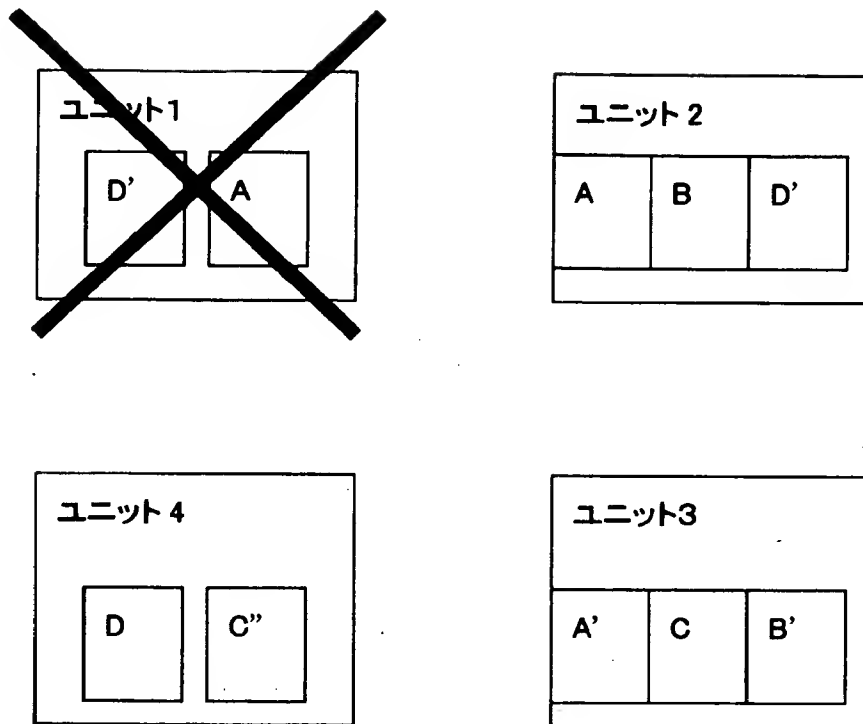
【図 1 9】



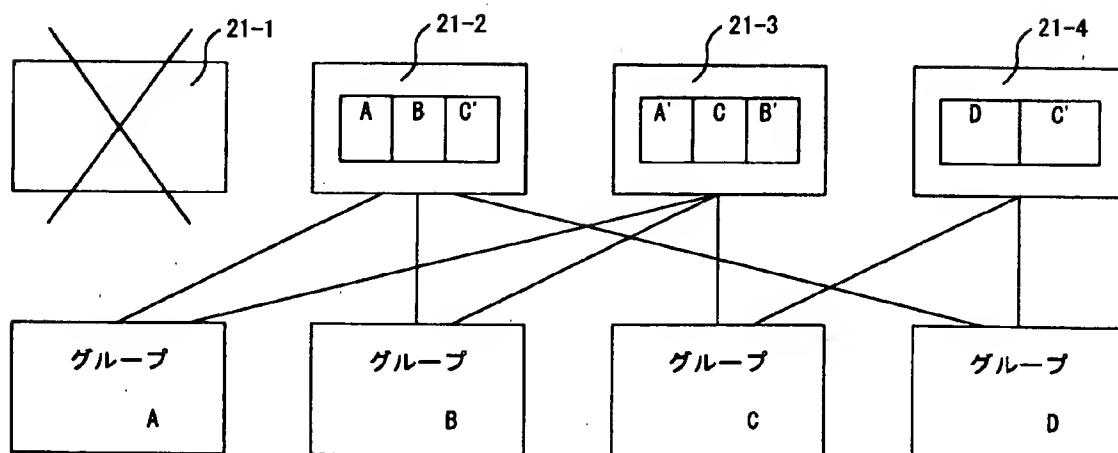
【図 2 0】

穴長グループ	ユニット			
	1	2	3	4
A	Active	Spare	No	No
B	No	Active	Spare	No
C	No	No	Active	Spare
D	Spare	No	No	Active

【図 2 1】



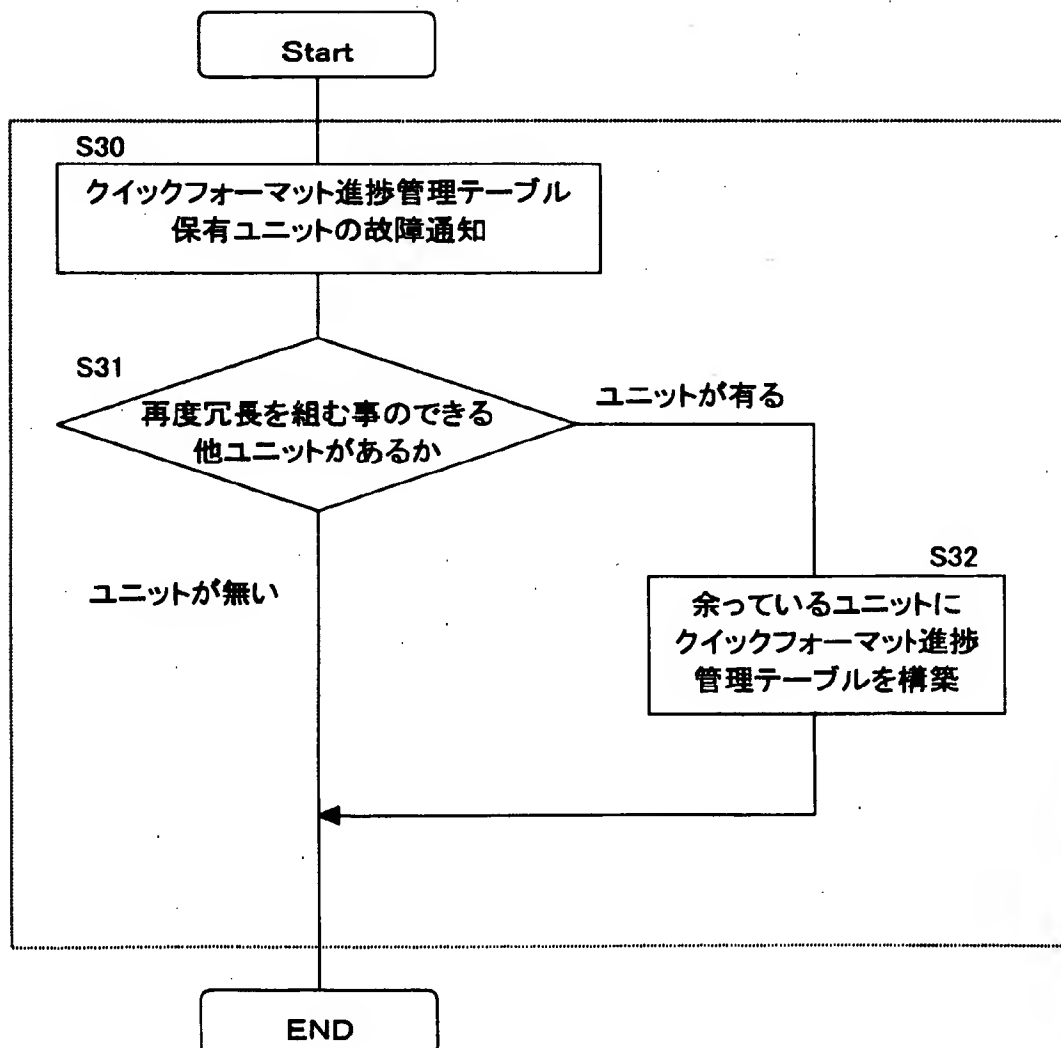
【図 2 2】



【図 23】

穴長グループ	ユニット			
	1	2	3	4
A	No	Active	Spare	No
B	No	Active	Spare	No
C	No	No	Active	Spare
D	No	Spare	No	Active

【図 24】



【書類名】 要約書

【要約】

【課題】 冗長構成をとる物理ディスク群に対し、ホストアクセスを可能とする論理フォーマットを行う R A I D 制御装置に関し、冗長構成を維持しつつ、ホストアクセスを待たせることなく、物理ディスクの論理フォーマットを実行する。

【解決手段】 冗長グループを組む複数の物理ディスク群（2）と、論理ボリュームの論理フォーマット処理の進捗情報を管理するための管理テーブル（33、34）を参照して、論理ボリュームの各領域の論理フォーマット要求を発行する複数の論理フォーマット管理モジュール（21-1～21-4）と、論理フォーマット管理モジュールの処理要求に応じて、前記物理ディスクを実アクセスする複数の下位層モジュール（22）とを有する。そして、前記各論理フォーマット管理モジュールは、前記各物理ディスク群の管理テーブルを有し、ディスクアクセス要求に応じて、管理テーブルを参照し、当該アクセス又は論理フォーマット要求かを判定し、下位層モジュールに要求を発行する。

【選択図】 図2

認定・付加情報

特許出願の番号	特願 2001-220179
受付番号	50101067552
書類名	特許願
担当官	濱谷 よし子 1614
作成日	平成13年 7月26日

<認定情報・付加情報>

【特許出願人】

【識別番号】	000005223
【住所又は居所】	神奈川県川崎市中原区上小田中4丁目1番1号
【氏名又は名称】	富士通株式会社

【代理人】

申請人

【識別番号】	100094514
【住所又は居所】	神奈川県横浜市港北区新横浜3-9-5 第三東 昇ビル3階 林・土井 国際特許事務所
【氏名又は名称】	林 恒徳

【代理人】

【識別番号】	100094525
【住所又は居所】	神奈川県横浜市港北区新横浜3-9-5 第三東 昇ビル3階 林・土井 国際特許事務所
【氏名又は名称】	土井 健二

出 願 人 履 歴 情 報

識別番号 [000005223]

1. 変更年月日 1996年 3月26日

[変更理由] 住所変更

住 所 神奈川県川崎市中原区上小田中4丁目1番1号

氏 名 富士通株式会社